OPTIMIZATION AND DECISION STRATEGIES FOR MEDICAL PREPAREDNESS AND EMERGENCY RESPONSE

A Thesis
Presented to
The Academic Faculty

by

Chien-Hung Chen

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
H. Milton Stewart School of Industrial and Systems Engineering

Georgia Institute of Technology
December 2013

OPTIMIZATION AND DECISION STRATEGIES FOR MEDICAL PREPAREDNESS AND EMERGENCY RESPONSE

Approved by:

Dr. Eva K Lee, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Renato D.C. Monteiro
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. David Goldberg
School of Industrial and Systems Engineering
Georgia Institute of Technology

Mr. Ferdinand Pietz
Division of Strategic National Stockpile
Centers for Disease Control and Prevention

Dr. Yajun Mei
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Hongyuan Zha
School of Computational Science and Engineering
College of Computing
Georgia Institute of Technology

Date Approved: 22 August 2013

To my parents

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my advisor, Dr. Eva Lee. I am eternally grateful for her patience and encouragement that guide me through this toughest doctoral program and complete my thesis. I learned a lot from her about how to be a good researcher, and what it really is to conduct excellent research. I would also like to thank Professor David Goldberg, Professor Yajun Mei, Professor Renato Monteiro, Mr. Ferdinand Pietz, and Professor Hongyuan Zha for serving on my thesis committee, and my heartfelt appreciation to their encouragement and support.

I gratefully acknowledge the funding sources that made my Ph.D. study possible. The work was partially supported by grants to Dr. Lee from the Centers for Disease Control and Prevention and from the National Science Foundation. Special thanks go to the Georgia Aquarium for providing invaluable collaboration opportunity.

Fellows from the Center for Operations Research in Medicine and HealthCare, Kyungduck Cha, Tsung-Lin Wu, Anna Yang, Niquelle Brown, Fan Yuan, and all other group members, it has been my honor and pleasure to work and learn with you on various projects. I thank all my friends in Atlanta from the bottom of my heart. Thank you for sharing part of your life with me. My life in Atlanta is not without struggle. Some of you taught me life can be imperfect, some of you taught me life can be colorful. It is all of your friendship that defines who I am today.

Lastly, I would like to express my deepest gratitude and love to my parents for their encouragement, tolerance, and love. I am the most fortunate person in the world to be their son.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The public health emergencies, such as bioterrorist attacks or pandemic outbreaks, have gained serious public and government attentions since the 2001 anthrax attacks and the SARS outbreak in 2003. These events require large-scale and timely dispensing of critical medical countermeasures for protection of the general population.

This thesis research focuses on developing mathematical models, real-time algorithms, and computerized decision support systems that enable **(**1) systematic coordination to tackle multifaceted nature of mass dispensing, (2) fast disease propagation module to allow immediate mitigation response to on-site uncertainties, and (3) user-friendly platform to facilitate modeling-solution integration and cross-domain collaboration. The work translates operations research methodologies into practical decision support tools for public health emergency professionals.

Under the framework of modeling and optimizing the public health infrastructure for biological and pandemic emergency responses, the task first determines adequate number of point-of-dispensing sites (POD), by placing them strategically for best possible population coverage. Individual POD layout design and associated staffing can thus be optimized to maximize throughput and/or minimize resource requirement for an input throughput. Mass dispensing creates a large influx of individuals to dispensing facilities, thus raising the risk of high degree of intra-facility infections. Our work characterizes the interaction between POD operations and disease propagation.

Specifically, fast genetic algorithm-based heuristics were developed for solving the integer-programming-based facility location instances. The approach has been applied to the metro-Atlanta area with a population of 5.2 million people spreading over 11 districts. Among the 2,904 instances, the state-of-the-art specialized integer programming solver

solved all except one instance to optimality within 300,000 CPU seconds and solved all except 5 to optimality within 40,000 CPU seconds. Our fast heuristic algorithm returns good feasible solutions that are within 8 percent to optimality in 15 minutes. This algorithm was embedded within an interactive web-based decision support system, RealOpt-Regional© . The system allows public health users to contour the region of interest and determine the network of PODs for their affected population. Along with the fast optimization engine, the system features geographical, demographical, and spatial visualization that facilitate real-time usage. The client-server architecture facilities front-end user interactive design on Google Maps© while the facility location mathematical instances are generated and solved in the back-end server.

In the analysis of disease propagation and mitigation strategies, we first extended the 6-stage ordinary differential equation-based (ODE) compartmental model to accommodate POD operations. This allows us to characterize the intra-facility infections of highly contagious diseases during local outbreak when large dispensing is in process. The disease propagation module was then implemented into the CDC-RealOpt-POD© discrete-event-simulation-optimization. CDC-RealOpt-POD is a widely used emergency response decision support system that includes simulation-optimization for determining optimal staffing and operations. We employed the CDC-RealOpt-POD environment to analyze the interactions between POD operations and disease parameters and identified effective mitigation strategies. The disease propagation module allows us to analyze the efficient frontier between operational efficiencies and intra-POD infections. Emergency response POD planners and epidemiologists can collaborate under the familiar CDC-RealOpt-POD environment, e.g., design the most efficient plan by designing and analyzing both POD operations and disease compartmental model in a unified platform. Corresponding problem instances are formed automatically by combining and transforming graphical inputs and numerical parameters from users.

To facilitate the operations of receiving, staging and storage (RSS) of medical countermeasures, we expanded the CDC-RealOpt-POD layout design functions by integrating it with the process flow. The resulting RSS system allows modeling of both system processes along with spatial constraints for optimal operations and process design. In addition, agent-based simulation was incorporated inside where integrated process flow and layout design allow analysis of crowd movement and congestion. We developed the hybrid agent behavior where individual agents make decision through system-defined process flow and autonomous discretion. The system was applied successfully to determine guest movement strategies for the new Georgia Aquarium Dolphin Tales exhibit. The goal was to enhance guest experience while mitigating overall congestion.

CHAPTER 1

INTRODUCTION

Large-scale dispensing of medical countermeasures is vital to protect the public against acts of terrorism, natural disasters, and other public health emergencies. In 2001, despite the limited death resulted in an anthrax bio-attack, over 32,000 people were potentially exposed to the biological agent and had to receive prophylactic medication. Large-scale dispensing is critically important for control of infectious disease outbreak, such as the 2003 SARS and pandemic influenza.

Since 2004, Cities Readiness Initiative (CRI) has been established in the nation's largest cities and metropolitan statistical areas where more than 50% of the U.S. population resides. One focus of CRI involves rapid response to potential aerosolized anthrax attack in a large metropolitan area. As a result, state, local and tribal public health departments were required to develop dispensing plans to protect the population in response to a large-scale bioterrorist. CRI highlights the importance of immediate response within the 48-hour time window after outbreak because of the steep mortality rate. In addition, it emphasizes the importance of last mile dispensing of medical countermeasures to the affected population.

Medical preparedness issues and strategies have been investigated in [21][25][40][47]. Human aspect of bio-terror preparedness and response is investigated in depth as well [12][17][52]. In addition to the applications of surveillance and information technologies to enhance preparedness against bio-terror attacks [7][78], many researchers [20][22][27][44][45][46][73] have studied modeling of emergency response. In particular, Centers for Disease Control and Prevention [9] and Mason and Washington [72] investigated optimal staffing arrangements when limited staffing is available. The

CDC study has proven to be invaluable because it offers insight on the practicality of a simulation system in emergencies. Giovachino and McCue [31] highlighted the importance of a real time system in which users can dynamically change the floorplan in response to patient flow. Lee et al, along with CDC, developed the RealOpt© system [58] that combines simulation and optimization into a real-time decision support environment [59][60][61]. The system offers strategic and tactical capabilities for optimizing staffing and layout design for individual dispensing sites [60][61][90]. The team also studied facility location to determine the best network dispensing facilities for best population protection [64]. Prior to this thesis research, RealOpt had over 3000 active public health user sites. In the rest of this thesis, we denote this CDC system as CDC-RealOpt-POD© .

As the continuation of Lee's collaborative work with CDC, this thesis research focuses on developing mathematical models, real-time algorithms, and computerized decision support systems that enable (1) systematic coordination to tackle multifaceted nature of mass dispensing, (2) fast disease propagation module to allow immediate mitigation response to on-site uncertainties, and (3) user-friendly platform to facilitate modeling-solution integration and cross-domain collaboration. The work aims to translate operations research methodologies into practical decision support tools for public health emergency professionals.

The first task determines adequate number of point-of-dispensing sites (POD), by placing them strategically for best possible population coverage. Individual POD layout design and associated staffing can be optimized via CDC-RealOpt-POD to maximize throughput and/or minimize resource requirement for an input throughput. Large-scale dispensing creates a large influx of individuals to dispensing facilities, thus raising the risk of high degree of intra-facility infections. Our work characterizes the interaction between POD operations and disease propagation.

Fast genetic algorithm-based heuristics were developed for solving the integer-programming-based facility location instances. The approach has been applied to the metro-Atlanta area with a population of 5.2 million people over 11 districts. Among the 2,904 instances, the state-of-the-art specialized integer programming solver solved all except one instance to optimality within 300,000 CPU seconds and solved all except 5 to optimality within 40,000 CPU seconds. The fast heuristic algorithm returns good feasible solutions that are within 8 percent to optimality in 15 minutes. This algorithm was embedded within an interactive web-based decision support system, RealOpt-Regional. The system allows public health users to contour the region of interest and determine the network of PODs for their affected population. Along the fast optimization engine, the system features geographical, demographical, and spatial visualization that facilitate real-time usage. The client-server architecture facilitates front-end user interactive design on Google Maps© while the facility location mathematical instances are generated and solved to near-optimality in the back-end compute server.

Secondly, in the analysis of disease propagation and mitigation strategies, we extended the 6-stage ordinary differential equation-based (ODE) compartmental model that combines POD operations to characterize the intra-facility infections of highly contagious diseases during local outbreak. This disease propagation module was then implemented into the CDC-RealOpt-POD discrete-event-simulation-optimization environment for modeling intra-facility disease propagation. CDC-RealOpt-POD is a widely used emergency response decision support system that includes simulation-optimization. We analyzed the interactions between POD operations and disease parameters and identified effective mitigation strategies. The disease propagation module allows us to analyze the efficient frontier between operational efficiencies and intra-POD infections. Emergency response POD planners and epidemiologists can collaborate under the familiar CDC-RealOpt-POD environment, e.g., to design the most efficient plan by

designing and analyzing both POD operations and disease compartmental model in a unified platform. Corresponding problem instances are formed automatically by combining and transforming graphical inputs from users into mathematical formulation.

Thirdly, to facilitate the operations of receiving, staging and storage (RSS) of medical countermeasures, we expanded the CDC-RealOpt-POD layout design functions by integrating it with the process flow. The resulting RSS system allows modeling of both system processes along with spatial constraints for optimal operations and process design. In addition, agent-based simulation was incorporated inside where integrated process flow and layout design allow analysis of crowd movement and congestion. We developed the hybrid agent behavior where individual agents make decision through system-defined process flow and autonomous discretion. The system was applied successfully to the Georgia Aquarium to analyze and design the Dolphin Tales exhibit's guest movement with the goal of enhancing the guest experience while mitigating overall congestion.

Figure 1 shows the web-based RealOpt-Regional. Figure 2 shows the integration of disease propagation and RSS modules within the CDC-RealOpt-POD environment.

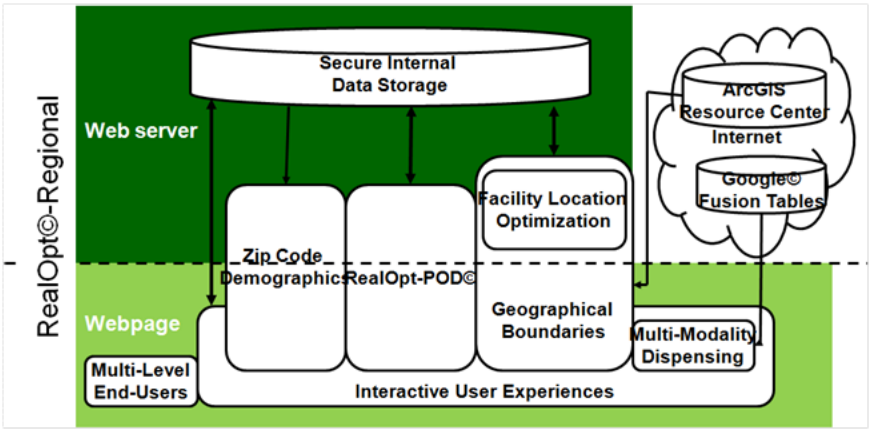


Figure 1 The web-based RealOpt-Regional environment for determining optimal network of dispensing sites

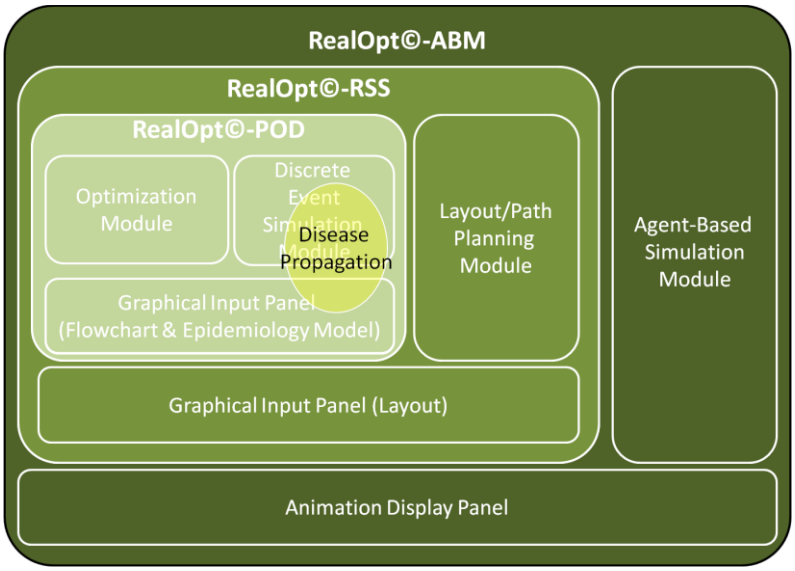


Figure 2 The integration of the disease propagation and RSS modules within the CDC-RealOpt-POD real-time simulation-optimization environment

CHAPTER 2

NETWORK OF DISPENSING FACILITIES

The writing and technical content of this chapter include excerpts and summary of published work [56][57][62][63][64][90]. Our contribution is presented in Section 2.3 and 2.5.

2.1 Background and Motivation

"Planning for a catastrophe involving a disease outbreak or mass casualties [3][5][83] is an ongoing challenge for first responders and emergency managers. They must make critical decisions on treatment distribution points, staffing levels, impacted populations and potential impact in a compressed window of time when seconds could mean life or death. Although extensive resources have been devoted to planning for a worst-case scenario on the local, regional and national scale, the U.S. Government Accountability Office found gaps still exist. While many states have made progress in planning for mass casualty events, many noted continued concerns related to maintaining adequate staffing levels and accessing other resources necessary to effectively respond."[90]

After outbreaks of potential public health emergencies, immediate and aggressive responses must be carried out. This includes large-scale medical countermeasures dispensing [45][87] to mitigate possible deaths and to control epidemic. Besides dispensing medication, "service constructs", as defined in [62][63], for food and water distribution, temporary shelters, medical care, screening and registry, and decontamination may also be required. Without loss of generality, we call such sites the point-of-dispensing sites (PODs). These are the places where individuals in the affected population come to receive services.

"While dispensing medical countermeasures or providing medical services require specific healthcare personnel, distribution of food and water or other personal needs can be carried out by other emergency workers and volunteers. Nonetheless, both types of services share key elements: resources are scare, time is precious, risk is uncertain and evolving, there is a large affected population to serve, and the on-the-ground conditions can be exceedingly stressful, for the impacted population and for the emergency workers. To maximize the throughput that can be served under limited resources of labor, time, and potential damaged infrastructure, PODs must be established such that they are flexible, scalable, sustainable, and agile for operations continuity and fluidity. Policy and decision makers must know where to establish such network of sites so as to impact the maximum overall outcome. Further, POD layout must be designed to facilitate the best operations workflow and throughput while minimizing resource requirements."[57]

2.2 Problem Formulation

For a given regional population and a geographical boundary, the problem is to determine the number of PODs to be opened and their locations so that the entire population of households in the region can be served in the most effective and efficient way. This problem was first studied and modeled via a two-stage mixed integer program approach [64]. Specifically, the number of PODs to be opened is firstly minimized; next the assignment of households to open PODs is determined such that the total weighted travel distance/time from households to PODs is minimized.

In this section, we first describe the formulation as introduced in [64]. This is followed by discussion of our rapid genetic algorithm-based heuristic for finding good feasible solutions [56] in Section 2.3.

"Let k be the total number of districts in the region, and let

G_i = set of cells in district i

$d(r,l)$ = distance between cells r and l

$d_{\max}$ = maximum allowed travel distance

c_l = the capacity of the facility at cell l.

p_r = population of cell r

Let the decision variable

y_l =1 if facility site at cell l is selected for setting up a dispensing facility, 0 otherwise

x_{rl} = 1 if the population in cell r is served by the facility at grid l, and 0 otherwise

The capacitated POD location problem can be formulated as follows:

(COVER-CAP) Minimize $\sum_{i=1}^{k} \sum_{l \in G_i} y_l$

subject to

$$\sum_{l \in G_i} y_l \geq 2, \forall i = 1,...,k \qquad (2\text{-}1)$$

$$d(r,l)x_{rl} \leq d_{\max} y_l, \forall r, l \in G_i, i = 1,...,k \qquad (2\text{-}2)$$

$$\sum_{l \in G_i} x_{rl} = 1, \forall r \in G_i, i = 1,...,k \qquad (2\text{-}3)$$

$$\sum_{r \in G_i} x_{rl} p_r \leq c_l, \forall l \in G_i, i = 1,...,k \qquad (2\text{-}4)$$

$$x_{rl}, y_l \in \{0,1\}, \forall r, l \in G_i, i = 1,...,k \qquad (2\text{-}5)$$

Constraint (2-1) ensures at least 2 POD locations are chosen. Constraints (2-2) ensure that each household will travel at most d_{max} miles, (2-3) ensures that every household is served, and (2-4) ensures that the capacity of the facility is not exceeded. Capacity constraints are needed, for example, when fire code restrictions stipulate the maximum number of people that can be inside a building or areas within a building. Capacity constraints are also important if there are parking limitations on the premises. (2-4) can be removed if no capacity restrictions are required." In this model, the targeted region is discretized into cells (one mile by one mile as discretization resolution). The population

of targeted region can be evenly distributed to each cell, or strategically placed based on census density of the population.

The solution of **COVER-CAP** is used in the second model **MINAVG-CAP.** Let

n_i = number of facilities in district *i*, as determined by **COVER-CAP**.

"MINAVG-CAP minimizes the distance traveled by all households, $\sum_{i=1}^{k}\sum_{r\in G_i}(\sum_{l\in G_i} x_{rl}d(r,l)p_r)$, while satisfying the constraints (2-2) – (2-5) above, as well as the constraints $\sum_{l\in G_i} y_l = n_i, \forall i = 1,...,k$, where n_i is the optimal number of PODs setup for district *i* returned from **COVER-CAP."** [64]

2.3 Genetic Algorithm-based Heuristics

[64] described a general-purposed mixed integer programming (MIP) solver used to solve these facility location instances, the instances have proven to be intractable for the most competitive commercial MIP solver.

Facility location problem (FLP) is a well-known difficult combinatorial optimization problem. There have been many variants of FLP studied extensively in the literature. Due to its combinatorial nature and NP-hard property, active research has been done in developing approximation algorithms. The following table summarizes the approximation factors of these algorithms [71].

Table 1 Approximation algorithms for uncapacitated facility location problems

Approximation Factor	Reference	Technique
O(logn)	Hochbaum [37]	Greedy algorithm
3.16	Shmoys et al. [85]	LP rounding
2.41	Guha and Khuller [33]	LP rounding + greedy augmentation
1.736	Chudak and Shmoys [11]	LP rounding
$5 + \varepsilon$	Korupolu et al.[50]	local search
3	Jain and Vazirani [43]	primal-dual method
1.853	Charikar and Guha [10]	primal-dual method + greedy augmentation
1.728	Charikar and Guha [10]	LP rounding + primal-dual method + greedy augmentation
1.861	Mahdian et al. [41][70]	greedy algorithm
1.61	Jain et al. [41][42]	greedy algorithm
1.582	Sviridenko [86]	LP rounding
1.52	Mahdian et al. [71]	greedy algorithm + cost scaling
1.50	Byrka [8]	LP rounding
1.488	Li [66]	LP rounding + greedy algorithm

The table illustrates the difficulty of these problems. By developing approximation algorithms that have polynomial time computation complexity, the current best approximation factor derived is 1.488.

Lee and Zaider derived hypergraphic cutting planes and incorporated them along with smart geometric heuristic and matrix reduction, and solved some intractable facility-like instances [65]. The 2,904 facility location instances arising from the Atlanta metropolitan network of point-of-dispensing determination range from 20,000 0/1 variables to over 10 million 0/1 variables. Lee and Zaider were able to solve all the instances except one to optimality within 300,000 CPU seconds. All the instances except five can be solved to

optimality within 40,000 CPU seconds. Among these 5 instances, the capacity limit is less than 500 per hour. Four of them required approximately 300,000 CPU seconds to solve to optimality, and the remaining one remains unsolved. In particular, among all the solved instances, a feasible solution that is within 5 percent of optimality is obtained within 5,000 seconds [56].

The difficulty in solving these instances exactly and the relatively big gap in the solutions obtained from the best approximation algorithm prompted us to look for good heuristic solutions. Specifically, in order to address the facility location problems for scenario-based analysis, we resort to meta-heuristics. Although there is usually no mathematical guarantee on the solution quality, they are widely used in solving difficult optimization problems with good practical and empirical performance.

To speed up the computational time for scenario-based analysis, we designed a specialized heuristic approach that couples together the features of a genetic algorithm [28] and an adaptive greedy search. When using the permutation representation in chromosome design, the challenge here is to find a set of *feasible* open facilities that satisfy both the capacity constraints and the distance constraints. In addition, *maintaining solution feasibility during the evolutionary process can require extra computational effort and may reduce solution quality*. To overcome this, we introduced the concept of potential served set for each candidate facility. This is analogous to Aickelin's set-covering approach [2]. Using potential served set in the decoding procedure, the solution feasibility becomes independent of the evolutionary process. It also provides a better opportunity to eliminate redundant open facilities.

Instead of randomly generating every chromosome, the initial population is partly generated via randomization and partly by the *k*-mean clustering algorithm to ensure better initial and final solution quality. Furthermore, kick-move and local search are applied to maintain the balance between search diversification and intensification.

Figure 3 depicts the computation scheme of our genetic algorithm-based heuristic designed for the facility location problems. Table 2 includes the list of algorithmic parameters.

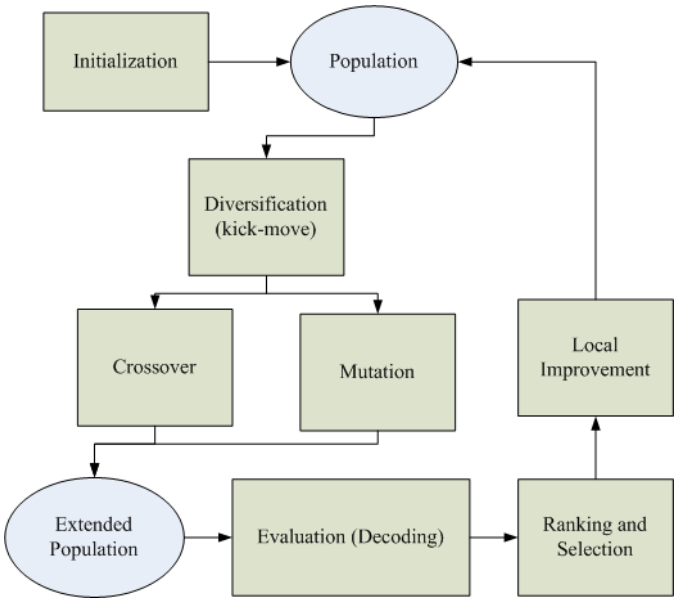


Figure 3 Computation scheme of a genetic algorithm-based heuristic for the facility location problem

Table 2 Genetic algorithm parameters

Pop	Population size.
Gen	Number of generation.
P_c	Crossover rate.
P_m	Mutation rate.
KM	Kick-move threshold.
P_g	Percentage of *k*-mean-initialized population.

2.3.1 Chromosome representation

The permutation representation, originally proposed for the traveling salesman problems (TSP) [28], is incorporated into our procedure to represent the location

sequences. Each gene in a chromosome represents the index of a corresponding cell. The length of each chromosome equals the number of cells to be covered. However, permutation representation is an indirect encoding schema, and thus the actual cells covering can be derived only after decoding. Figure 4 illustrates the relationship between chromosome representation and its corresponding discretized targeted region.

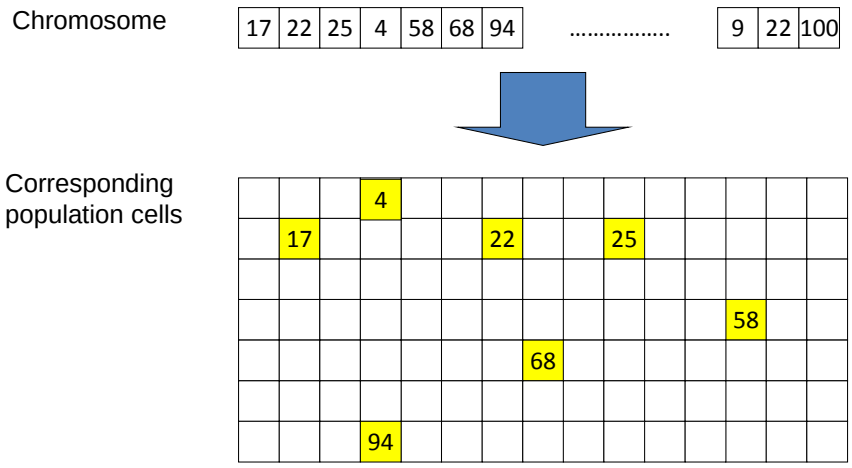


Figure 4 Conversion from chromosome representation to corresponding targeted cells

2.3.2 Initialization

The initial population is generated in two parts: randomly generated, and *k*-mean-initialized. Let P_g denote the percentage of *k*-mean-initialized population. In other words, there are $Pop * P_g$ initial chromosomes generated by the *k*-mean clustering algorithm.

We applied the *k*-mean clustering algorithm to get an initial greedy solution and corresponding chromosomes. We define LBd= $\frac{\text{Total Demand}}{\text{Facility Capacity}}$, and LBa= $\frac{\text{Size of Total Area}}{\text{Size of Facility Coverage}}$. Initially, LB=max{LBd, LBa} (intuitive lower bound for number of facilities) seeds are randomly generated and considered as open facilities to

generate a feasible solution by the *k*-mean clustering algorithm. If these open facilities are not enough to generate a feasible solution due to the distance constraints or the capacity constraints, LB is incremented, seeds are randomly generated, and the *k*-mean clustering algorithm is applied again until a feasible solution is successfully constructed.

To construct an initial chromosome from an initial solution, we assign the indices of the opened facilities in the initial solution to the beginning of the chromosome in random order, and fill the rest of the chromosome with the remaining indices in random order. Hence, each *k*-mean generated solution corresponds to one (but could be multiple) chromosome in the initial population.

2.3.3 Genetic Operators

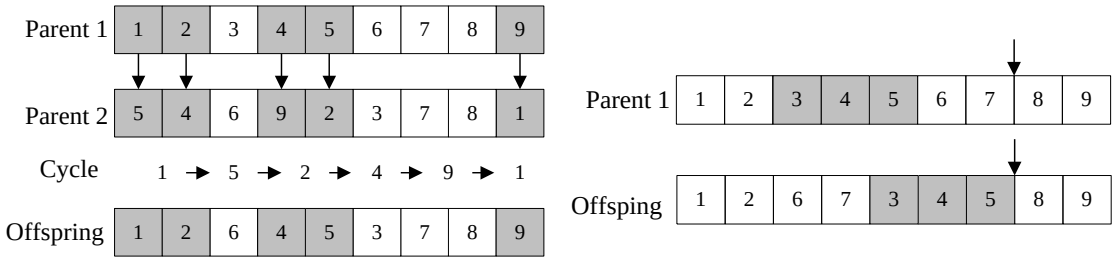
During the evolution process, cycle crossover and displacement mutation are adopted as the genetic operators to generate offspring from the parents to preserve chromosome legality and feasibility. The procedures of cycle crossover and displacement mutation, as illustrated in Figure 5 (a) and Figure 5 (b), are summarized as follows:

Cycle Crossover:

- Find the cycle that is defined by the corresponding positions of genes between parents.
- Copy the genes in the cycle to a child with the corresponding positions of one parent.
- Determine the remaining genes for the child by deleting those genes that are already in the cycle from the other parent.
- Fulfill the child with the remaining genes.

Displacement Mutation:

- Select a segment at random and insert it back at a random position.



(a) Cycle crossover (b) Displacement mutation

Figure 5 Illustration of genetic operators

2.3.4 Decoding Procedure

The decoding procedure is used to transform the sequence of indices in the chromosome into a feasible solution for the facility location problem. This solution has to satisfy both the capacity constraints and the travel distance constraints.

Instead of using the indices on chromosome as the sequence of opening facilities and then assign cells to their closest and feasible facility, we generate a potential served set for each cell (potential facility). By doing so, indices on chromosome are now considered as the sequence of potential served sets. *This maintains the solution feasibility independently of the evolution process.* It also provides us with a better opportunity to eliminate redundant opened facilities.

Each cell has only one potential served set, but can belong to multiple potential served sets. Decoding by using potential served sets can guarantee generating only feasible solution. Figure 6 graphically illustrates the concept of potential served sets.

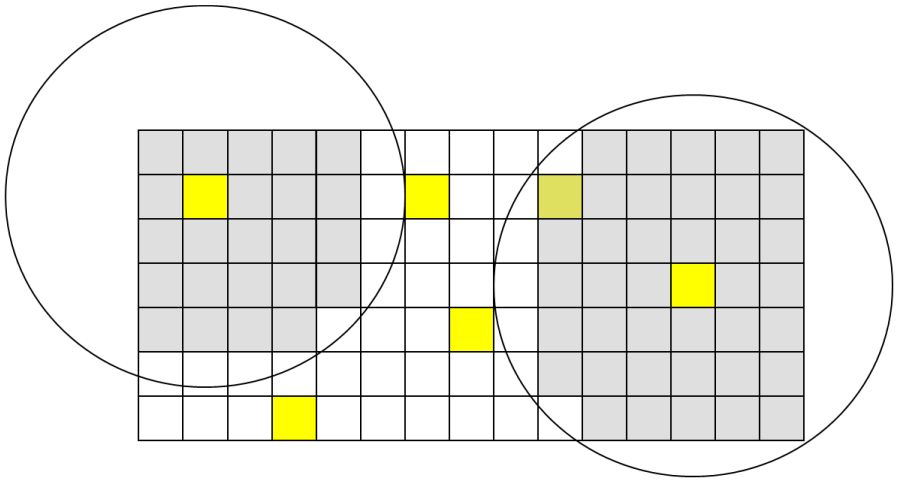


Figure 6 Illustration of potential served sets

In the decoding procedure, redundancy removal is conducted in a heuristic manner to keep the decoding procedure computationally tractable.

After removing redundant served sets, cell is assigned to its closest open facility. It is possible for a cell to be assigned to multiple served sets before being assigned to an open facility. It is unnecessary to check the assignment feasibility since potential served sets are all feasibly generated.

2.3.5 Ranking and Selection Mechanism

After decoding, the fitness value of each chromosome is calculated as the weighted sum of number of PODs opened and the total distance travelled. The weights are chosen such that the number of PODs opened always dominates the total travel distance.

The roulette wheel approach was adopted to select the new population. Instead of using the original fitness value, the selection probability is defined as follows:

$$p_k = \frac{2k}{Pop(Pop+1)}$$

where k refers to the k^{th} chromosome in descending order of fitness value, and p_k denotes the selection probability of the k^{th} chromosome. This selection probability can prevent premature convergence and maintain the distinguishability between the chromosomes in the late stage of the entire evolution process.

2.3.6 Diversification (kick move)

By experimentation we found that, when using the genetic algorithm on solving the facility location instances, the solution converges quickly and is readily trapped in the local optimal. The kick-move function is added to diversify the searching possibility.

Kick-move is applied at the beginning of each generation. Given that the kick-move threshold (KM) is defined as the percentage of population with the same number of facilities opened, the kick-move replaces all additional chromosomes with the same number of facilities with randomly-generated sequences.

2.3.7 Local Improvement

The number of facilities is determined after removing redundant potential served sets. However, local improvement is still possible to further improve the weighted average distance by modifying the exact location of facilities and cell's assignment after decoding. Local improvement incorporates several local search procedures to improve the current solution.

Insertion

In insertion (Figure 7), cell *r* is taken away from cluster *i* and put into cluster *j* if this can contribute to improved weighted average distance. Insertion is conducted under both the distance constraints and the capacity constraints.

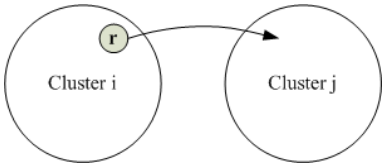


Figure 7 Illustration of local improvement—insertion

Switch

In switch (Figure 8), while cell r_1 is taken away from cluster *i* and feasibly inserted into cluster *k*, simultaneously cell r_2 is taken away from cluster *j* and feasibly inserted into cluster *i*, if this can contribute to improved weighted average distance.

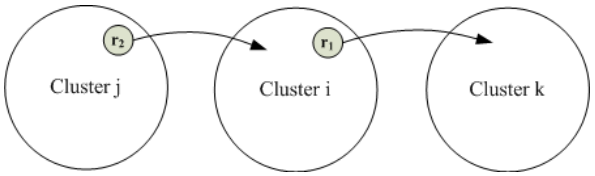


Figure 8 Illustration of local improvement—switch

Ideally, applying local improvement to all solutions can lead to a better solution. However, due to the computation intensity of these local search procedures, especially Switch, we only applied them to the best solution in each generation.

2.4 Computational Performance

Both the mathematical programming approach and the genetic algorithm-based heuristic are applied to the metro Atlanta area for numerical comparison. The population is about 5.2 million spreading over 28 counties that are represented by 11 districts. Figure 9 shows the map of these 11 districts and their associated county(ies).

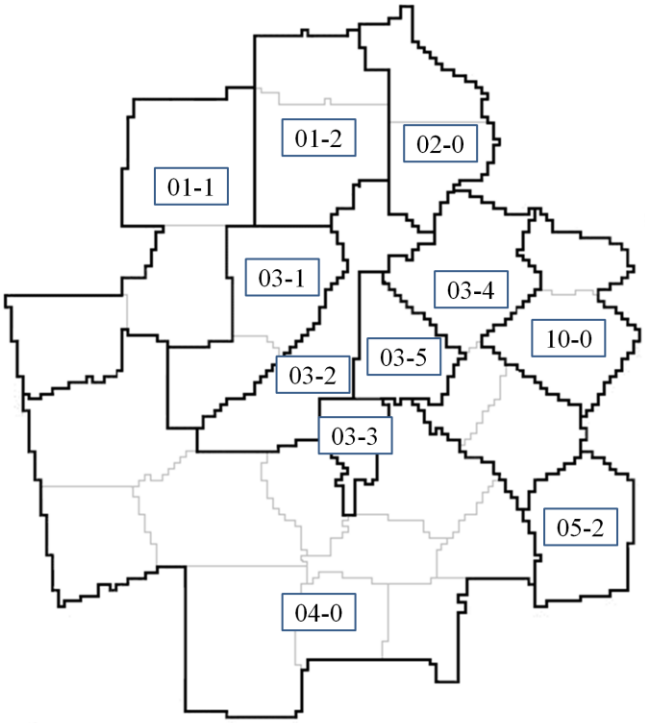


Figure 9 Map of 11 districts and corresponding counties in metro Atlanta area

Table 3 summarizes the population in each county.

Table 3 Demographic information of metro Atlanta area by counties

District Index	County	Population
01-1	Bartow	91,226
	Haralson	28,616
	Paulding	121,530
01-2	Cherokee	195,327
	Pickens	29,640
02-0	Dawson	20,643
	Forsyth	150,968
03-1	Cobb	679,325
	Douglas	119,557
03-2	Fulton	960,009
03-3	Clayton	271,240
03-4	Gwinnett	757,104
	Rockdale	80,332
	Newton	91,451
03-5	DeKalb	723,602
04-0	Henry	178,033
	Fayette	106,671
	Spalding	62,185
	Coweta	115,291
	Carroll	107,325
	Butts	23,561
	Lamar	16,679
	Pike	16,801
	Meriwether	22,881
	Heard	11,472
05-2	Jasper	13,624
10-0	Barrow	63,702
	Walton	79,388

By discretizing each district at 1-mile by 1-mile resolution, the number of cells per district ranges between 36 and 3275, and the number of households within each cell ranges between 140 and 3074. Figure 10 compares the number of binary variables used in the mathematical model for each of the 11 districts. Only two instances have fewer than 100,000 binary variables.

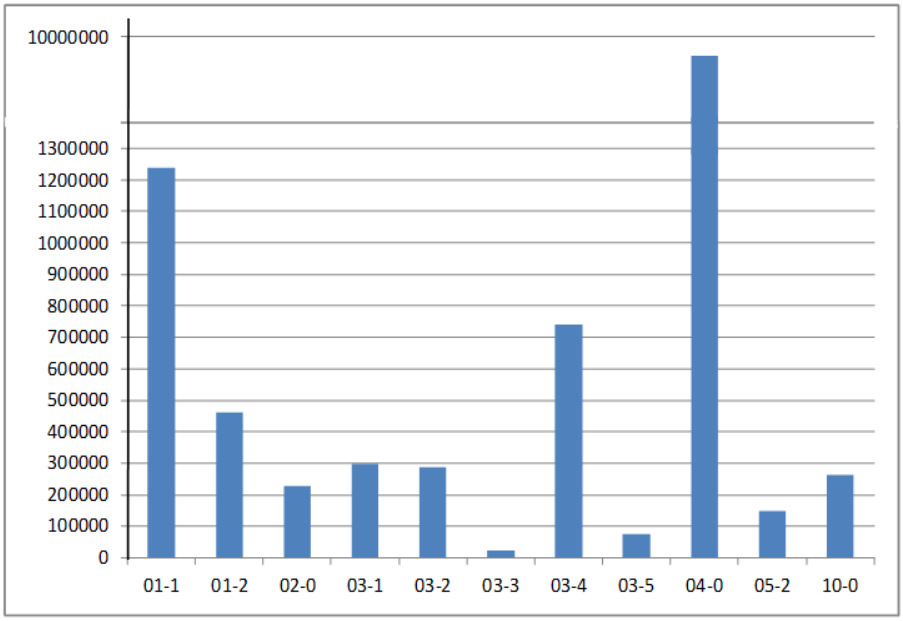


Figure 10 Number of binary variables in each POD-location MIP instances for 11 districts [56]

The benchmark tests were performed on these instances using CPLEX V11. CPLEX returned an optimal number of facilities for the smallest instance within 30 CPU minutes when the POD capacity was set to 2000 individuals per hour. However, for this same instance, it cannot solve the problem within a week of CPU time when the facility capacity is 500 per hour. CPLEX was not able to solve any of the other instances after running for several months of CPU time.

The challenge was to find optimal or near-optimal solutions rapidly to enable decision process. Using recent computation advances for solving intractable facility-like instances [65], all the instances except one can be solved to optimality within 300,000 CPU seconds. All the instances except five can be solved to optimality within 40,000 CPU seconds. Among these 5 instances, the capacity limit is 500 per hour. Four of them required approximately 300,000 CPU seconds to solve to optimality, and the remaining

one remains unsolved. In particular, among all the solved instances, a feasible solution that is within 5 percent of optimality is obtained within 5,000 seconds [56].

Figure 11 contrast the optimal solutions with the results obtained using our genetic algorithm approach. We vary the capacity of the facilities and the maximum allowed travel distance and gauze their impact on the number of facilities opened.

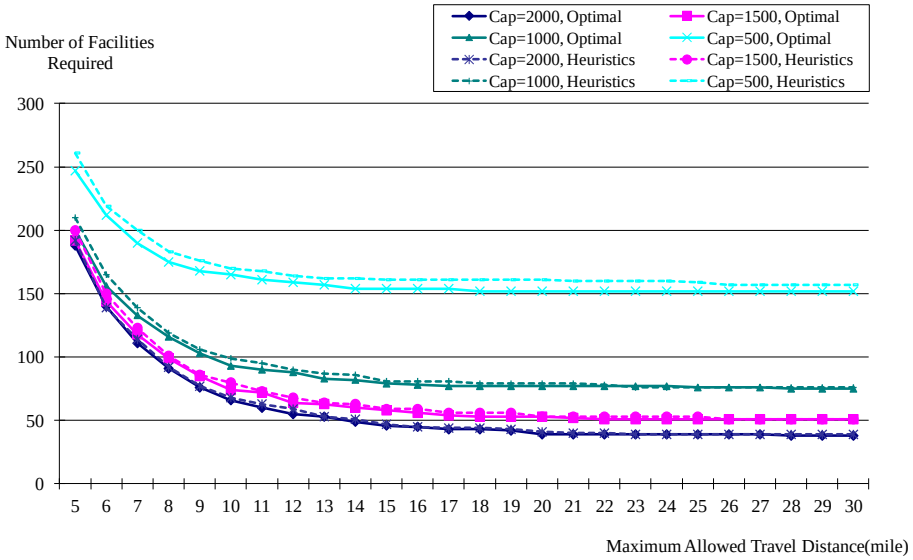


Figure 11 Number of facilities required under different maximum allowed travel distance (mile) when capacity limit is 2000, 1500, 1000, and 500 individuals per hour [56]

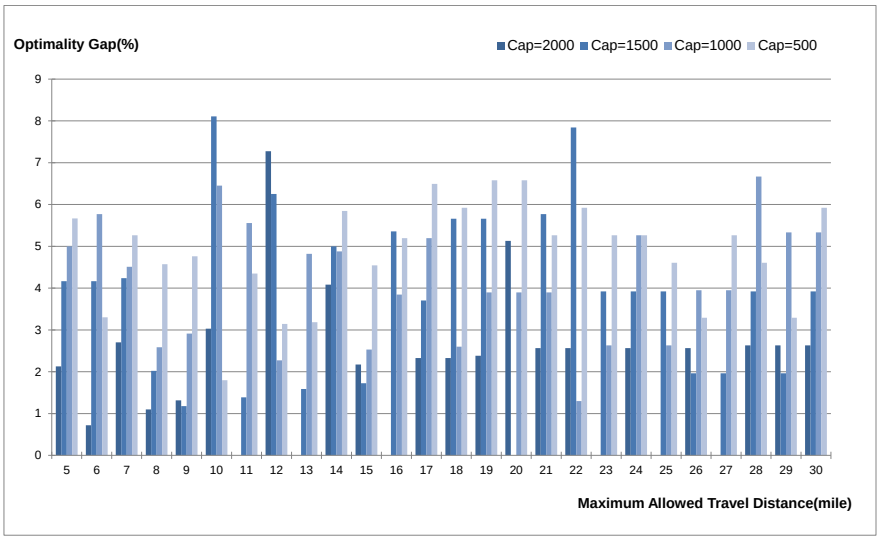


Figure 12 Optimality gap under different maximum allowed travel distances (mile) and capacity limits (individuals per hour)

The rapid solution engine and quality of returned solutions facilitate the study of efficient frontiers to analyze the trade-offs in determining the most suitable number of strategically placed POD sites that best serve the targeted region. For the metro Atlanta area, our POD-location solver allows us to obtain good feasible solutions that are within 8 percent from optimality in less than 15 minutes (The hardest two instances with 500-capacity constraints ran for about 15 minutes; most other instances required less than 3 minutes). Figure 13 shows the distribution of distances traveled by household for a solution with a capacity limit of 1500 per hour and a maximum travel distance set to 10 miles.

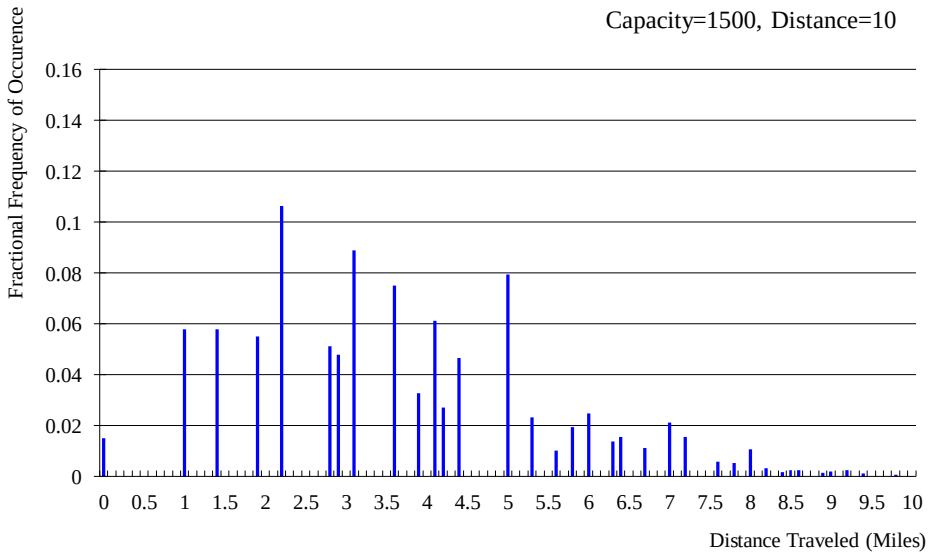


Figure 13 Actual distance that each household travels to its assigned POD when the facility capacity is 1500 and travel distance limit is 10 miles [56]

2.5 A Real-Time Decision Support System: RealOpt-Regional

RealOpt-Regional© is an interactive online software enterprise for large-scale regional medical dispensing and emergency preparedness. It features interactive visualization tools to assist with spatial understanding of important landmarks in the region, assess the population densities and demographic makeup of the region, and identify potential facility locations. It also features backend mathematical models for large-scale facility location problems [64] and a backend novel and rapid solution engine (Section 2.3) for strategic and operational planning and real-time dynamic optimization.

RealOpt-Regional has been available to public health and emergency planners since December 2009. In 2010 there were two webinars sponsored by CDC's Division of Strategic National Stockpile for broadcasting RealOpt-Regional. Currently there are over 200 regional leaders using RealOpt-Regional, most of the users are regional program directors or state project coordinators in the Departments of Public Health or Departments of Emergency Management.

Much of the material in this section consist of excerpts from the submitted paper [57] and published articles [62][63].

We first describe the system architecture and design of RealOpt-Regional. Next, the system functionalities are presented, beginning with functionalities that relate to the user experience, and moving on to functionalities that pertain to managing geographical boundaries; facility location and population assignment; ZIP code and population composition; and multi-modality dispensing indicators. Through secured login, users can access to CDC-RealOpt-POD system for POD layout design and resource allocation, and bio-surveillance.

The system provides a unified platform for functional integration of data collection, data analysis, and decision support.

2.5.1 System Architecture and Design

RealOpt-Regional is implemented via the client-server architecture. On the client-side, the Google Maps JavaScript API [91] along with JavaScript are incorporated inside Java Server Pages (JSP) to provide the primary I/O functionalities. Using Tomcat 6.0.24 as the web server, the system application is written in JSP so as to integrate with Java-based extension.

The rapid heuristic algorithm, described in Section 2.3, is implemented in C++ as the solution engine for determining the near-optimal dispensing facility network. Java Native Interface (JNI) is employed to link the Java and C++ program on the server to complete the backend computation and interface. By taking the solution engine as a native method and connecting to JSP through JNI, the solution engine can be implemented in C, C++, Matlab, or even Python to leverage their computational advantages and well-developed

numerical libraries. The system architecture is designed so as not to modify any component except the junction module when extending the solution engine.

The intensive optimization computation is performed on the server and requires no extra computing capacity from the user. User's computer is primarily used for webpage display and object rendering on the map. Asynchronous JavaScript and XML (AJAX) are employed in our system for connecting the server computation and client I/O interactively. AJAX is used for sending and receiving requests between server and client in an asynchronous manner. The asynchronous connection in RealOpt-Regional is critical in the following manner: Since it usually takes time to initiate and solve the facility location problem on the back-end, AJAX is implemented to send optimization request to the server so that users do not have to wait for response from the server during optimization. Another crucial advantage is that users can still retrieve optimization results even if they are disconnected from the server. (Potential internet disruption may occur due to various reasons during the running of the optimization process. While the connection can be manually or automatically re-built, the running progress and solutions will not be accessible unless another request is sent from client web browser to our server.) By using AJAX, users can still get either updated running progress or optimization results when the program runs to completion.

A schematic design of the RealOpt-Regional system architecture is depicted in Figure 14.

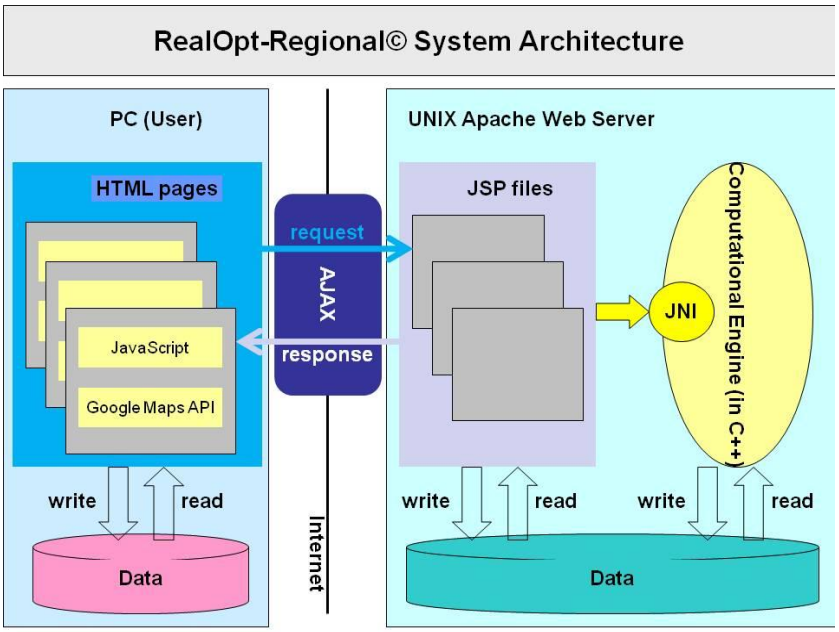


Figure 14 System architecture of RealOpt-Regional

The system has been migrated from Google Maps JavaScript API V2 [92] to Google Maps JavaScript API V3 since 2011. This ensures system compatibility with new features such as Google fusion tables and Styled Maps.

2.5.2 System Functionalities

Figure 15 shows the interface of RealOpt-Regional for public health emergency preparedness users. On the left is a Google map, and on the right are five function panels: Jurisdiction, POD(Marker), Facility Location Optimization, Draw Polygon, and Local Search. Help files and practice scenarios are provided in the bottom-right to help users learn the various functionalities.

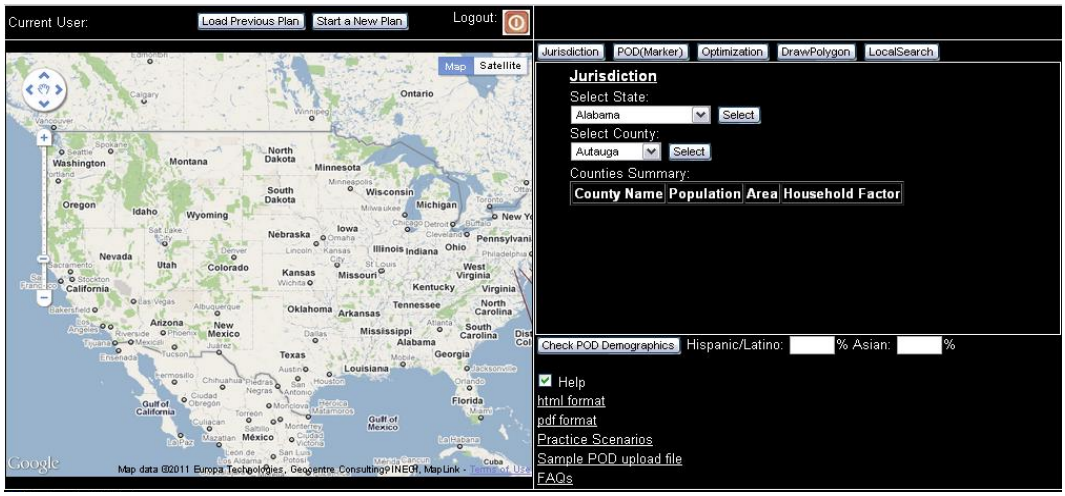


Figure 15 System interface for public health emergency preparedness directors and planners [57]

With system flexibility and expansion in mind, we designed this system in modular form (Figure 16) and allowed direct linkage to functional modules. Currently, it is composed of twelve primary modules for integrating system interactivity, client-server architecture, optimization, and decision support. We briefly describe each module below.

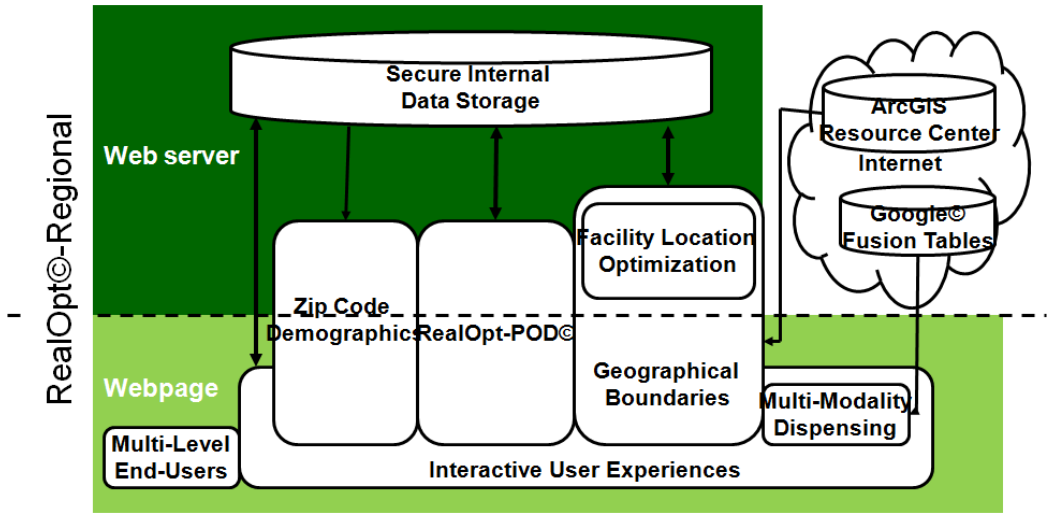


Figure 16 System modules of RealOpt-Regional

Interactive User Experience We incorporate the graphical interactivity enabled by Google Maps JavaScript API V3 as part of the I/O functionalities. This equips users with spatial understanding of the planning and surrounding regions. Also, without the need for prerequisite knowledge in mathematical programming, users are empowered to build, optimize, and evaluate dispensing networks in a visualization environment. They can interact with the map by selecting region of interest, adding/removing potential PODs (as markers), defining dispensing constraints (capacity constraints at each POD site, and travel distance constraints), and setting household factor (the average number of occupants per household in a jurisdiction). All these information are sent to the server for building a mathematical model and for optimization.

Geographical Boundaries By linking to ArcGIS [93], RealOpt-Regional can display the geographical county boundaries of the United States. Emergency response users can select and define their planning area spatially. Demographics such as population/density of each county are provided to facilitate building and optimizing the dispensing networks.

City boundaries can be defined by the free drawing polygon tool. This enables our system to work compatibly with the Cities Readiness Initiative to enhance preparedness in the nation's largest cities and metropolitan statistical areas. The flexibility of users manually drawing planning areas also ensures that the heterogeneity of population densities between cities and counties can be considered in building the dispensing networks. Further, such free drawing capability is critical for dealing with international sites (e.g., after the 2010 earthquake in Haiti) where affected regions are not pre-determined [57][62][63]. The drawing process, which includes adding vertex, removing vertex, and modifying the polygon shape, entails only a series of clicking and dragging operations on the map. Figure 17 illustrates this system capability with the drawn city boundary of the city of Los Angeles.

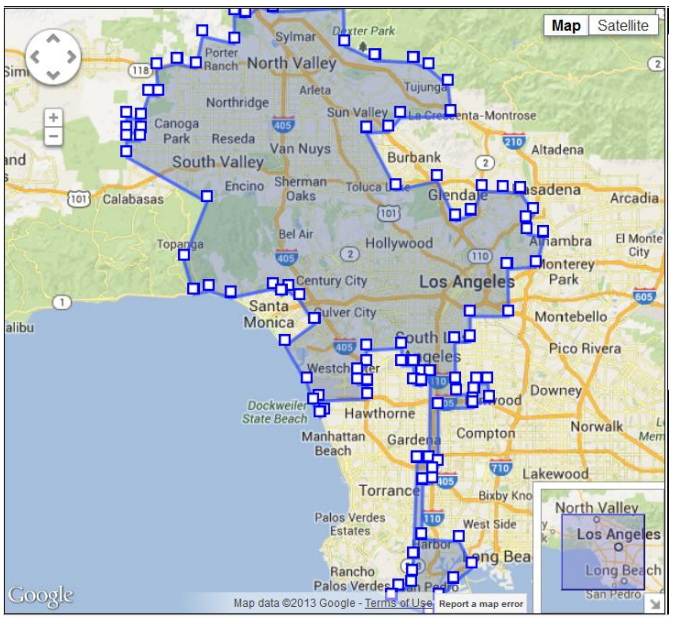


Figure 17 User-defined city boundary of the City of Los Angeles [57]

Facility Location and Population Assignment [56][64] Upon selecting the planning region, users can choose one of the following ways to input potential POD locations onto the map at their convenience. They can either input the addresses (or the name of that location) directly, upload a file with addresses, or interactively drag-and-drop the POD markers onto the map, where a targeted region is defined. Physical constraints on the facility must be modeled, e.g., capacity (number of households a facility can accommodate simultaneously during a certain period of time, this is often dictated by fire codes and other facility regulations). Users can specify the travel distance, household factors, and time for completion of dispensing can also be input.

Various objectives are incorporated within the RealOpt-Regional computational engines. In the event of catastrophic incidents, it is critical that PODs are strategically located so as to allow easy access by the affected public. Hence, minimizing transportation distance is one critical objective. Further, the setup and operating costs of

PODs cannot be neglected. In our model, setup costs and transportation distance form the composite objective criteria.

Once the potential PODs are selected and parameters for optimization are specified by users, this information is automatically translated into a facility location formulation in the back-end for optimization. Users can fetch the current feasible solution by manually interrupting the computation, or they can wait for the final solution when the solution engine terminates. Our implementation ensures that computation will not be interrupted even when the internet connectivity is disrupted.

ZIP Code and Population Composition By rearranging and analyzing zip code demographics and area boundaries [101][102], the system maintains a set of up-to-date census data bank that contains 32,036 postal-code areas in the United States. The system provides zip code based information including regional demographics, social indicators, economic indicators, and geographical boundary of postal-code areas.

After determining a set of optimal POD locations and population assignment, users can query POD coverage-based demographics by overlapping the POD service area with all adjacent zip code areas, and aggregating zip code-based data by considering (a) size of overlapping, and (b) population density of each overlapped zip code area.

The importance of POD coverage based demographics is that, while the demographics based on zip code area or county is static and can be available through other sources, the POD coverage based demographics is completely dynamic and dependent on user-defined parameters in optimizing facility locations. Using this information, emergency planners can identify appropriate personnel combination for each POD individually.

Figure 18 illustrates that by overlapping POD coverage with all adjacent zip code areas, we can identify demographics of the majority population served by the POD.

Different icons reflecting demographics can then be labeled on each POD accordingly, as shown in Figure 19.

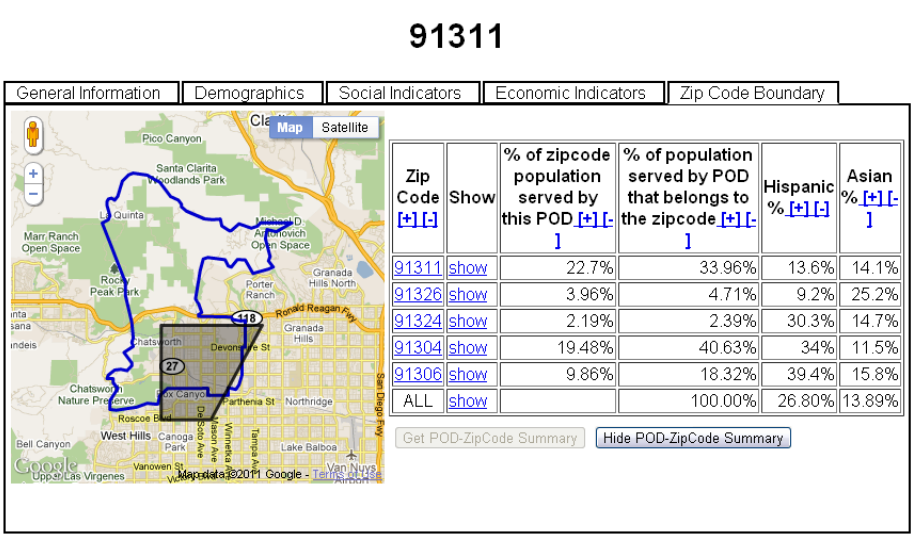


Figure 18 POD demographics for 10010 De Soto Ave, Chatsworth, CA 91311[57][62]

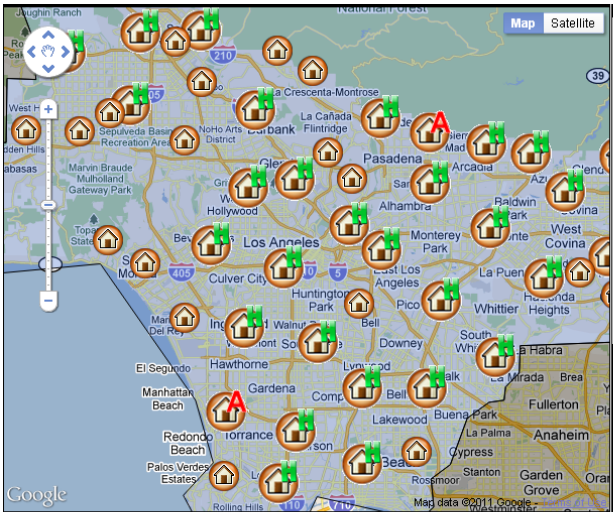


Figure 19 Difference between POD Demographics. “H” stands for “Hispanic/Latino”, and “A” stands for “Asian” [57][62]

Multi-Modalty Dispensing Indicator The key to mass dispensing is to protect the general population efficiently and effectively under time pressure. For example, in an anthrax attack, the goal is for citizens to receive antibiotic prophylaxis within 48 hours of the determination that an attack has occurred, as the mortality rate for persons demonstrating symptoms of inhalation anthrax is extremely high. Thus, it is recognized that multiple dispensing modalities often must be employed in order to serve (cover) the entire regional population [56][64].

Open PODs can be drive-through or facility-based walk-through. Special needs populations may need closed PODs on site. For example, nursing homes, assisted living facilities, homeless shelters, hospitals, and prisons house many residents for whom it would be inconvenient or inadvisable to travel to a public dispensing facility.

Further, self-organized closed PODs should be promoted. Corporate offices that staff a large number of employees could be served in a similar manner. Once these sites receive prophylactic supplies, they could set up a closed POD within their building, with their own healthcare staff and volunteers, or with public health staff supplemented by the state. Several factors suggest that such closed PODs will have fewer security concerns and will be easier to manage than public PODs. These factors include familiarity with the environment and people (e.g., fellow residents/employees), existing security measures including established checkpoints and previously authenticated identification badges with photo and/or biometric markers, and less stress than having to commute to a public POD.

Airports and hotels, where a large number of non-resident travelers can be found, are also candidates for setting up PODs. This is especially important with infectious disease where travelers will be protected first before they leave for elsewhere. Universities can use their own health facilities (and if necessary, additional mobile on-campus PODs provided by the state) to provide prophylaxis to on-campus students, staff and faculty. Clearly, if large employers and medical facilities provide prophylaxis to their own

employees, families, and patients, it will eliminate a high percentage of the population (may be as high as 40% in some large cities) from visiting public PODs, thus reducing the load on those facilities.

POD markers in RealOpt-Regional can distinguish drive-through mode and walk-through mode, and with specific demographics needs. Currently, the system maintains a data bank of 2,000 homeless shelters [99], 4,200 universities and colleges [97], 5,128 airports [95], 12,881 high schools [94][98], 73,079 assisted living facilities [96], and 24,192 hotels [100]. Public health and emergency response planners are equipped with this information and can make better decisions in constructing more cost-effective and efficient mass dispensing networks. Users can also search for specific business or facility types to explore options of dispensing sites.

Multi-Level End-Users RealOpt-Regional can be used by both emergency preparedness personnel and the general public to obtain dispensing site information. It supports multiple levels of authorization control access.

2.6 Conclusions

In this chapter, we introduced a novel interactive web-based public health informatics decision support system, RealOpt-Regional, for medical countermeasures dispensing, emergency preparedness planning, and bio-surveillance. Coupled with the newest Google Maps JavaScript API V3, the system provides public health emergency preparedness regional planners spatial understanding of the planning region, empowers them to build a mathematical model for optimizing a network of dispensing sites in a graphical visualization environment, and, more importantly, solve the problem in real time. The client-server architecture not only relieves the need for intensive computation from the system users, it also embodies the concept of system modularization and encapsulation so

as to maintain the extendibility of the solution engine without interfering with the user experience.

CHAPTER 3

DISEASE PROPAGATION ANALYSIS AND MITIGATION STRATEGIES FOR EFFECTIVE MASS DISPENSING

3.1 Introduction

Considering possible terrorist attacks such as dissemination of anthrax spores, intentional food product contamination, and release of chemical weapons in major metropolitan subway systems [3][5][83], the threat level of terrorist attack has been dramatically increased in recent years. Moreover, the persistent outbreak of avian influenza in Asia since 1997 has caused public panic. There is serious concern regarding widespread pandemic as a result of person-to-person transmission of infectious diseases. When an outbreak occurs, whether it is a bioterrorist attack or a naturally occurring pandemic, it is crucial to launch immediate and aggressive public health emergency response such as mass dispensing of medical countermeasures [45][87] to decrease the possible death toll and to control the potential disease spread. Successful implementations can help reduce resources requirement, shorten patient waiting time, and maximize throughput.

Lee et al. first collaborated with the Centers for Disease Control and Prevention and developed the original CDC-RealOpt-POD system in 2004 [59][60][61][62]. The system intertwines simulation with optimization seamlessly into a real-time decision support system where optimization output feeds into simulation for operations statistics output and subsequent local search improvement on determining the best resource allocation for maximizing the throughput or minimizing resource usage given a targeted throughput.

The simulation-optimization iterates until a near-optimal solution is obtained. The optimization module involves a nonlinear mixed integer program that focuses on staff allocation where different types and skills of personnel being assigned to appropriate tasks.

The original CDC-RealOpt-POD system consists of local search heuristics as well as a fluid model optimization engine. These optimization engines are incorporated within the simulation environment to form the Java-based standalone decision support system. As expected, simulation with iterative optimization via local search heuristics offers more accurate operations performance characterization. Optimizing over each POD design and resource allocation with under a million throughput per shift per POD takes only seconds of computational time. However, computationally, it can be prohibitive (if one wants to design a single super-POD serving over tens of millions of people in a single shift).The fluid model estimates resource assignment rapidly and can lessen the computational burden of potentially super-size instances.

By 2007, there were over 3,000 CDC-RealOpt-POD public health user sites using the system in designing emergency preparedness plans and performing actual operations for responses to hurricanes, seasonal flu shots, fire, as well as full-scale anthrax emergency drills.

Our effort herein builds on top of the existing CDC-RealOpt-POD capabilities. The major contribution summarized in Chapter 3 includes:

- Extending the nonlinear mixed integer programming resource allocation to include equipment resources;
- Combining the heuristic and fluid model for the simulation-optimization framework into a hybrid warm-start fluid-model heuristics approach;

- Porting the theoretical novel 6-stage ODE disease propagation model onto the computational simulation-optimization environment;
- Extending the disease propagation to a general form for any stage by developing the fundamental mathematical framework.

3.2 Optimization of Point-of-Dispensing (POD) Operations

3.2.1 Minimize Resource Allocation and Maximize Throughput: Previous Work

The optimization of multiple resources allocation involves placement of different resource groups at various stations to maximize throughput or minimize resource needs while satisfying a predetermined throughput. We follow the same notation as the original nonlinear mixed integer programming resource allocation [59][60][61][62]. Constraints in the model include:

- Maximum limits on wait time and queue length;
- Assignability and availability, for each resource group, of resource types at each station;
- Maximum limit on the cycle time of the individual; and
- Range of utilization desired at each station;

The model parameters are as follows:

$\mathbf{R}$ – the set of resource groups.

$\mathbf{T}_r$ – the set of resource types in resource group r, $r \in \mathbf{R}$

$\mathbf{S}$ – the set of processes in the process flow.

$\mathbf{S}_{ir} \subseteq \mathbf{S}$ – the set of processes on which resource type *i* in resource group *r* can be assigned. This models the assignability.

k_{ijr} – the cost of assigning a resource type *i* in resource group *r* to process *j*. $r \in \mathbf{R}, i \in \mathbf{T}_r,\ j \in \mathbf{S}_{ir}$.

$\overline{m_{ijr}}$ and $\underline{m_{ijr}}$ – the maximum and minimum number of resource type *i* in resource group *r* that can be assigned to process *j*. $r \in \mathbf{R}, i \in \mathbf{T}_r,\ j \in \mathbf{S}_{ir}$.

n_{ir} – the number of available resource type *i* in resource group *r*. $r \in \mathbf{R}, i \in \mathbf{T}_r$.

w_j, q_j, and u_j – the average wait time, average queue length, and average utilization rate, respectively, at process *j*. $j \in \mathbf{S}$.

C – the average cycle time (i.e., the length of time a customer spends in the system)

θ – the average throughput (number of customers served in a specified period)

Let the decision variable $x_{ijr} \in \mathbf{Z}_+$ be the number of resource type *i* in resource group r assigned to station j. $r \in \mathbf{R}, i \in \mathbf{T}_r,\ j \in \mathbf{S}_{ir}$.

We can represent the cost at each station j as $g_j\left(\sum_{(i,r)\in\mathbf{\Omega}_j} k_{ijr}x_{ijr}, w_j, q_j, u_j\right), j \in \mathbf{S}$, where $\mathbf{\Omega}_j = \{(i,r)|r \in \mathbf{R}, i \in \mathbf{T}_r, j \in \mathbf{S}_{ir}\}$. The total system cost depends on the cost at each station, and on system parameters, such as cycle time and throughput. Thus we can represent the total cost as $f\left(\sum_{j\in\mathbf{S}} g_j, c, \theta\right)$. Here, g_j and f are functions that are not necessarily expressible in closed form. We can formulate a general representation of the multiple resources allocation problem as

$$\begin{aligned}
&min \quad z = f\left(\textstyle\sum_{j\in\mathbf{S}} g_j, c, \theta\right) && && (0)\\
&s.t. \quad \underline{m_{ijr}} \le x_{ijr} \le \overline{m_{ijr}}, && \forall\, r \in \mathbf{R}, i \in \mathbf{T}_r,\ j \in \mathbf{S}_{ir} && (1)\\
&\quad\ \textstyle\sum_{j\in\mathbf{S}_{ir}} x_{ijr} \le n_{ir}, && \forall\, r \in \mathbf{R}, i \in \mathbf{T}_r && (2)\\
&\quad\ w(x)_j \le w_{max} && \forall\, j \in \mathbf{S} && (3)
\end{aligned}$$

$$q(x)_j \le q_{max}$$
$$u_{min} \le u(x)_j \le u_{max}$$
$$\theta(x) \ge \theta_{min}$$
$$c(x) \le c_{max} \quad (4)$$
$$x_{ijr} \in \mathbf{Z}_+ \qquad \forall\, r \in \mathbf{R}, i \in \mathbf{T}_r,\ j \in \mathbf{S}_{ir} \quad (5)$$

Constraints (0) to (5) form a nonlinear mixed integer programming problem for cost minimization under the constraints of multiple resources allocation and stochastic system performance. These are nonlinear MIP models, and are generally very difficult to solve (See Nemhauser and Wolsey [76]).

3.2.2 Solution Engine

As described in [59][60][61], CDC-RealOpt-POD utilizes a hybrid heuristic algorithm that is a cross between a greedy algorithm and a local search. In this heuristic, the first stage is used for initialization of the system. It works by greedily assigning workers to various stations within the POD. The assignment is based on initial estimates on queue length, which are in turn based on the existing probability density functions at each station. Once all the workers have been assigned this way, the second stage takes over. The algorithm observes and registers the formation of queues with various lengths at different stations, and quickly and dynamically performs local search to remove violations of system statistics in an attempt to improve the objective value. The implementation of the algorithm is guided strongly by the layout design, worker types, and links between different work-stations within the facility.

This heuristic routine is ad-hoc, and function evaluation is performed via simulation, and thus can be computationally very expensive. Figure 20 shows the algorithmic schema of this heuristic algorithm that has been tested and shown to offer a fast and robust

solution engine within the simulation process. It can return optimal staffing within seconds for a POD with a million throughput per shift per POD.

In addition to the heuristic algorithm, CDC-RealOpt-POD also includes a fluid model that estimates analytically staffing need [62].

Once the worker allocation is determined, a max-flow min-cost network problem is solved to determine worker assignment based on their respective skills set and priorities.

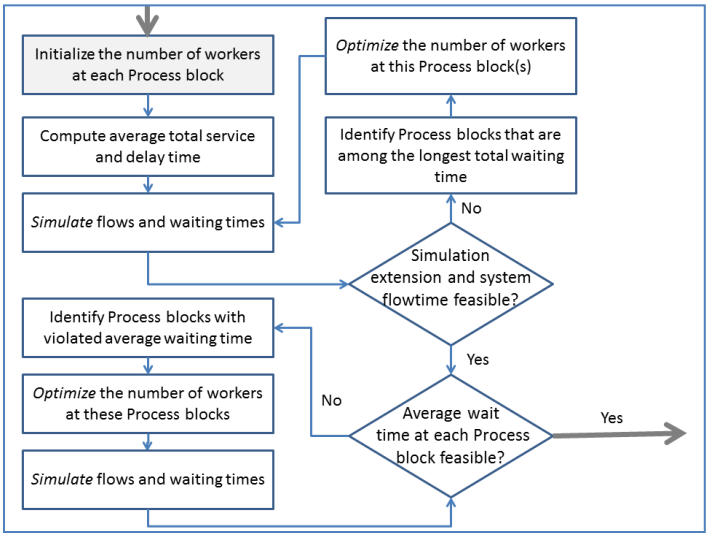


Figure 20 Solution engine for the original CDC-RealOpt-POD [59][60][61][62]

In this thesis work, we seek to improve the computation in Figure 20 by linking the fluid model as a warm-start with the heuristic algorithm to tackle the multiple resource allocation problems. This also enhances the computational capability for the 6-stage disease propagation simulation (Section 3.3).

A fluid model depicts entities flow along the network continuously, like fluid. There is a rich body of literature on fluid models. Some fundamental and influential advocates of the fluid view to queueing systems include Newell [77] and Hall [34][35]. It has been shown to be a good approximation of network flows of discrete units, and the continuous

nature of it enables us to obtain analytical, closed-form results. By noticing that the real entity nature (usually customers or patients) is in fact discrete, the gap between real discrete nature and approximate continuous nature can then be addressed by discrete event simulation.

Specifically, we design to solve this problem in two parts. First, we determine the multiple resources allocation by using estimated system performance and use the fluid model to return the initial assignment. Taking the results of multiple resources allocation as a warm-start, next, a detailed discrete-event simulation is launched to simulate the exact system performance by taking system stochasticity in consideration. Knowing that the estimated system performance such as waiting time, queue length, utilization and cycle time could be under-estimated and throughput could be over-estimated, if the system performance constraints are still feasible after running the simulation, the solution process is complete; otherwise, resources are incrementally increased in selected process in the process flow and system performance will be checked again by running simulation until system performance constraints are all met.

In multiple resources allocation, each resource group can be considered separately when system performance is estimated. However, resources from different groups need to be synchronized to be considered available when discrete event simulation is performed. This increases the potential infeasibilities when running simulation by using estimated resources allocation. More iteration, along with longer computation time, is expected this part of our algorithm.

When optimizing POD operations, we can use these system performance metrics to check the feasibility of our solution under different constraints. Constraints considered in this study include (a) extension for completion time; (b) flow time; (c) waiting time; (d) queue length; and (e) utilization. None of the constraints among (a) to (e) is dominant.

Resource Allocation through Hybrid Method—Combine both Fluid Model and Simulation

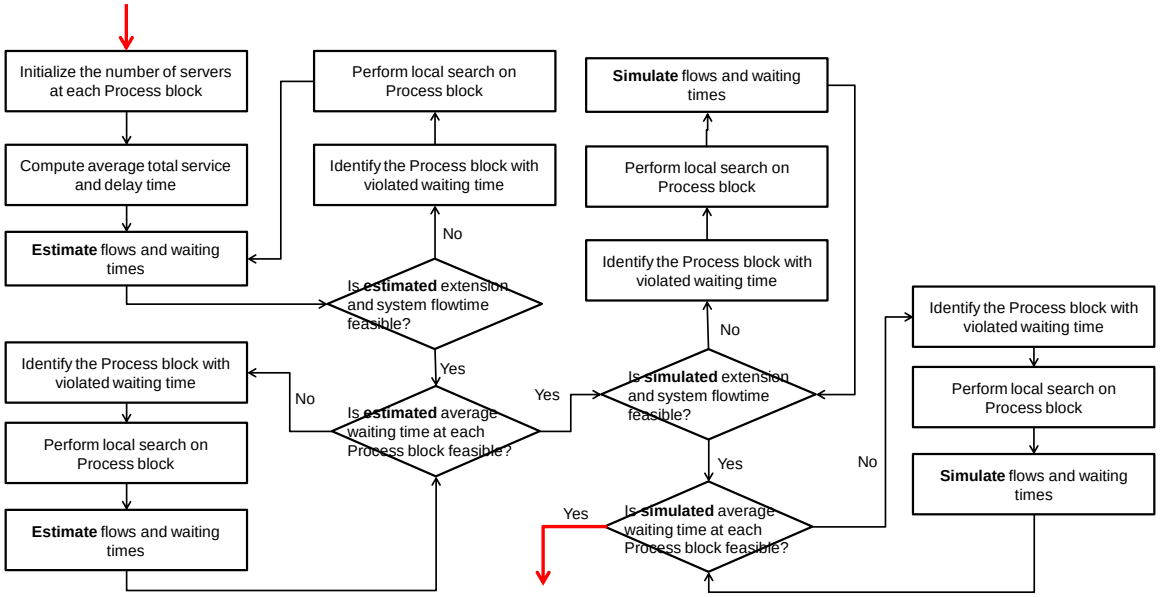


Figure 21 Local search-based optimization through hybrid of fluid model and discrete event simulation

3.3 Disease Propagation and Mitigation Strategies for Effective Mass Dispensing

The content of this section consists of excerpts from published papers [55][56][62].

Large-scale dispensing of medical countermeasures has been proven to be an effective way to contain the outbreak of highly infectious disease, such as smallpox, pneumonic plague, and pandemic flu. On the operational side, successful implementations can also help reduce resources requirement, shorten patient waiting time, and maximize throughput. However, large influx of individuals into the dispensing centers raises the potential risk of high degree of intra-facility cross-infections. Thus

thorough understanding about how the disease is propagated during the dispensing under different floor plan design and POD operations is necessary.

In this section, we take advantage of the simulation-optimization environment of CDC-RealOpt-POD and analyze the propagation of highly infectious disease within dispensing sites. As validation, a mathematical model based on ordinary differential equations is also developed for benchmarking the simulation results. We contrast the merits between analytical tools and simulation tools. Experimental results for base case study and sensitivity analysis will be presented to depict the dynamics of intra-POD disease propagation, and to explore feasible mitigation strategies for effective mass dispensing.

3.3.1 Background and Motivation

Numerous research has shown that social distancing, isolation, and quarantining can be effective in reducing infections during a pandemic outbreak [23][24][26][89]. When large-scale dispensing sites are set up for dissemination of vaccines and/or other medications, these clinics may facilitate the spread of the disease and hence raises the potential risk of high degree of intra-facility cross-infections due to high-volume of patient flow. Hence, thorough understanding of how the disease is propagated within the site during dispensing is crucial. Important factors could include floor plan layout, process operations, and the pandemic characteristics.

Numerous researchers have studied disease spreading across one or several open geographical regions [14][15][23][24][26][29]. In this work, we concentrate on disease propagation inside mass dispensing clinics. Considering that POD medical dispensing is essentially an open queueing network, we analyze the interaction between the propagation of disease and the operations of open queueing network. While optimizing

POD operations in terms of resources requirement, patient waiting times, and throughput in itself is mathematically challenging, analyzing the propagation of disease along with POD operations further increase the problem complexity and computation burden. We employ the efficient simulator-optimizer decision support system that CDC-RealOpt-POD offered to tackle the computationally-intensive disease propagation analysis. The short computational time empowers POD designers, managers, and epidemiologists to perform real-time analysis to meet the changing conditions during the dispensing processes.

This section is organized as follows: Section 3.3.2 shows the mathematical model for outer-POD and intra-POD disease propagation. This model incorporates specific characteristics that distinguish the disease spread in mass-dispensing clinics from other types of epidemiology models. This ODE model also serves as the benchmark to the large-scale simulation-optimization system. In section 3.3.3 we discuss a large-scale simulation/optimization system that is capable of analyzing the interaction between dispensing operations and disease propagation inside the mass-dispensing clinic. Section 3.3.4 shows the numerical results of flu propagation in a vaccination clinic. Lastly, section 3.3.5 concludes with discussions and future research direction.

3.3.2 Compartmental Model for Disease Propagation

This section describes the disease propagation model based on a system of ordinary differential equations (ODE) [4][16]. The model is designed to characterize the distinct features in a mass dispensing clinic.

3.3.2.1 Point-of-Dispensing

POD operations usually consist of a series of pre-determined service stations. Each individual thus needs to receive specific services in required stations. Treatment or service can be provided to an individual or to a group of a certain size. The latter is usually called batch service. (e.g. orientation or bus transportation)

Figure 22 shows a design of a dispensing site that consists of 5 stations: triage, registration, medical screening, special medical care, and drug dispensing.

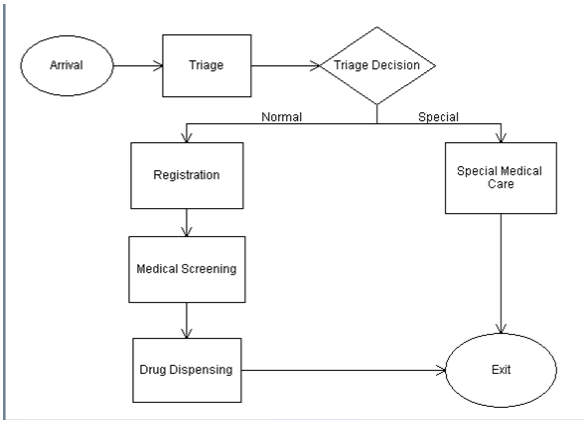


Figure 22 Example of a POD design [56]

3.3.2.2 Force of Infection

Infection force that is well-studied in the literature is primarily independent of the population size. Infection rate, β, denotes the average number of infection (or contact that is effective to incur transmission) of an infectious individual per unit time [36]. This parameter can be modeled to vary over time, but in most studies it is assumed to be a constant. Often times it is derived from basic reproduction number (R_0) and infectious period.

However, we note that infection force defined above does not apply well for POD study. The reason lies in its independence of population size. While the independence of infection force is, in fact, a better formulation for the disease spreading across an open

geographical region [4], our observation shows the necessity to consider the dependence of infection force on population size in mass-dispensing clinic, especially when treatment or service are provided to a group, as shown in Figure 27. Validation and detailed explanation will be provided in section 3.3.5.

In our analysis, we use the contact based transmission and incorporate two contact parameters γ and δ to represent outer-POD contact number and contact coefficient, respectively. This helps to model the infection force dependent on the population size. For clarification, the population size inside medical dispensing facility means the number of people in one station. People in one station include those who are being served and those who are waiting in the queue. Note that workers in that station are not included. The assumption in this analysis includes:

- Transmission only occurs among patients, and
- Transmission only occurs inside each station.

A contact is deemed effective if it leads to disease transmission. Under this definition the contact does not necessary need to be physical contact. The model is applicable to diseases transmitted by droplet contact transmission or airborne transmission. Outer-POD contact number means the average number of people an infectious/asymptomatic/symptomatic individual can contact outside the POD. Contact coefficient refers to the average number of infection (or effective contact) per contact per unit time.

Note that these parameters should be considered as the aggregation of demographic factors, sanitary measures, and disease essences. For example, while higher average population density means γ will also be higher, airborne transmission also leads to higher γ than droplet contact transmission since it has longer radius of transmission. In

addition, it is understandable that δ can be determined by the disease natural virulence and sanitary measures such as wearing masks.

Based on the definition of contact parameters it explains why the independence of infection force can formulate the disease spread across an open geographical region. The reason lies in that the population density in an open geographical area is usually assumed as a constant. The assumption suffices to show equivalence of infection rate-based transmission and contact based transmission under some circumstances. However, this assumption does not hold during a 36-hour mass dispensing campaign, especially when batch process station is included in the floorplan.

3.3.2.3 Six-stages Compartmental Model

Unlike the traditional SEIR model, in this study we use the novel SEPAIR 6-stages model [55][67][89] to capture the disease development, where S means susceptible stage, E means exposed stage, P means infectious stage, A means asymptomatic stage, I means symptomatic stage, R means recovered stage, and D mean died stage. Figure 23 shows the precedence relations between stages in SEPAIR model.

Two more disease stages, i.e. asymptomatic (A) and symptomatic (I), are incorporated into this model. People in infectious stage, asymptomatic stage, or symptomatic stage are capable of spreading diseases. The importance of using SEPAIR 6-stages model is that it provides the opportunities to examine further the interaction between POD designs, particularly the effect of triage accuracy, and disease spread.

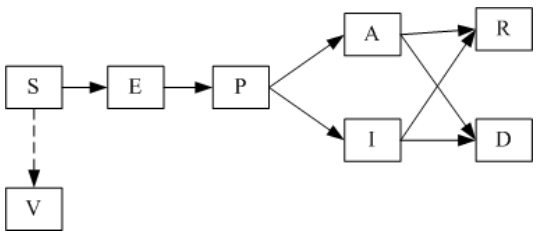


Figure 23 6-stages SEPAIR compartmental model [55]

If the POD is designed to vaccinate individuals, the stage representing being vaccinated needs to be included into the model in the corresponding station(s) to represent the change of individual susceptibility. In this study it is assumed that vaccines are only effective on susceptible individuals and can provide immediate protection.

While the disease propagation within POD operations is the primary focus of this study, propagation occurring before receiving services also plays an influential role. Due to the heterogeneous system behaviors inside and outside the POD, we introduce intra-POD disease propagation and outer-POD disease propagation to represent these system behaviors respectively within the compartmental model [55]. *This is the first mathematical model that integrates POD operations with disease spread analysis.*

3.3.2.4 Outer-POD Disease Propagation

We use the subscript "0" to indicate number of people (under different disease stages) outside POD. N_0, is used to denote population size, it means the number of living individuals. More specifically, we define the disease state space as $\Phi = \{S, E, P, A, I, R, D, V\}$, and outer-POD population size $N_0 = \sum_{\vartheta \in \Phi \setminus \{D\}} \vartheta_0$, where ϑ_0 is the number of outer-POD individuals of disease stage ϑ.

The constant p_S represents the probability of symptoms development, and p_D is the mortality probability. μ_E is the rate from exposed stage to infectious stage, μ_P is the rate from infectious stage to (a)symptomatic stage, μ_I (μ_A) is the rate from (a)symptomatic stage to recovered stage. r_a represents the constant arrival rate.

The following equations represent the rates of change of the population disease stages outside a POD:

- Susceptible stage

$$\frac{d}{dt}S_0 = -r_a \frac{S_0}{N_0} - \min\{N_0, \gamma\} \cdot \delta \cdot (P_0 + A_0 + I_0) \frac{S_0}{N_0} \quad (3\text{-}1)$$

- Exposed stage

$$\frac{d}{dt}E_0 = -r_a \frac{E_0}{N_0} - \mu_E E_0 + \min\{N_0, \gamma\} \cdot \delta \cdot (P_0 + A_0 + I_0) \frac{S_0}{N_0} \quad (3\text{-}2)$$

- Infectious stage

$$\frac{d}{dt}P_0 = -r_a \frac{P_0}{N_0} - \mu_P P_0 + \mu_E E_0 \quad (3\text{-}3)$$

- Asymptomatic stage

$$\frac{d}{dt}A_0 = -r_a \frac{A_0}{N_0} - \mu_A A_0 + (1 - p_S)\mu_P P_0 \quad (3\text{-}4)$$

- Symptomatic stage

$$\frac{d}{dt}I_0 = -r_a \frac{I_0}{N_0} - \mu_I I_0 + p_S \mu_P P_0 \quad (3\text{-}5)$$

- Recovered stage

$$\frac{d}{dt}R_0 = -r_a \frac{R_0}{N_0} + (1 - p_D)(\mu_A A_0 + \mu_I I_0) \quad (3\text{-}6)$$

- Dead stage

$$\frac{d}{dt}D_0 = p_D(\mu_A A_0 + \mu_I I_0) \quad (3\text{-}7)$$

3.3.2.5 Intra-POD Disease Propagation

For intra-POD disease propagation, we assume that there are K stations. We use subscript "*i*", *i* = {1, 2, ….., K}, to indicate different stations, each station is manned by n_i workers and each worker provides the same service at an identical nominal service rate λ_i. The population size of each station $N_i = \sum_{\vartheta \in \Phi \setminus \{D\}} \vartheta_i$, where ϑ_i is the number of individuals of disease stage ϑ at station *i*. In addition, we define the function

$h_i(\vartheta) = \lambda_i \min\{n_i, N_i\}\frac{\vartheta_i}{N_i}$, where $\vartheta \in \Phi \setminus \{D\}$, and i={1, 2, ….., K}. $h_i(\vartheta)$ can be interpreted as the real service rate of individuals of disease stage ϑ at station i.

The following equations (3-8) to (3-15) represent the rates of change of the population disease stages inside the POD:

- Susceptible stage

$$\frac{d}{dt}S_i = \psi_i r_a \frac{S_0}{N_0} + \sum_{j\in\{1,2,\cdots K\}}(1-\phi_{ji})q_{ji}h_j(S) - h_i(S) - \delta\cdot(P_i + A_i + I_i)\cdot S_i \quad (3\text{-}8)$$

- Exposed stage

$$\frac{d}{dt}E_i = \psi_i r_a \frac{E_0}{N_0} + \sum_{j\in\{1,2,\cdots K\}}q_{ji}h_j(E) - h_i(E) - \mu_E E_i + \delta\cdot(P_i + A_i + I_i)\cdot S_i \quad (3\text{-}9)$$

- Infectious stage

$$\frac{d}{dt}P_i = \psi_i r_a \frac{P_0}{N_0} + \sum_{j\in\{1,2,\cdots K\}}q_{ji}h_j(P) - h_i(P) - \mu_P P_i + \mu_E E_i \quad (3\text{-}10)$$

- Asymptomatic stage

$$\frac{d}{dt}A_i = \psi_i r_a \frac{A_0}{N_0} + \sum_{j\in\{1,2,\cdots K\}}q_{ji}h_j(A) - h_i(A) - \mu_A A_i + (1-p_S)\mu_P P_i \quad (3\text{-}11)$$

- Symptomatic stage

$$\frac{d}{dt}I_i = \psi_i r_a \frac{I_0}{N_0} + \sum_{j\in\{1,2,\cdots K\}}(1-\eta_{ji})q_{ji}h_j(I) - h_i(I) - \mu_I I_i + p_S\mu_P P_i \quad (3\text{-}12)$$

- Recovered stage

$$\frac{d}{dt}R_i = \psi_i r_a \frac{R_0}{N_0} + \sum_{j\in\{1,2,\cdots K\}}q_{ji}h_j(R) - h_i(R) + (1-p_D)(\mu_A A_i + \mu_I I_i) \quad (3\text{-}13)$$

- Dead stage

$$\frac{d}{dt}D_i = p_D(\mu_A A_i + \mu_I I_i) \quad (3\text{-}14)$$

- Vaccinated stage

$$\frac{d}{dt}V_i = \sum_{j\in\{1,2,\cdots K\}}\phi_{ji}q_{ji}h_j(S) + \sum_{j\in\{1,2,\cdots K\}}q_{ji}h_j(V) - h_i(V) \quad (3\text{-}15)$$

Three binary constants are used to represent the operational flow inside the POD: ψ_i =1 means station i is the first station and 0 otherwise. ϕ_{ji} =1 means station j is vaccination station and station i is its direct downstream station and 0 otherwise. q_{ji} is used to model the transition percentage from station j to station i. Lastly, η_{ji} denotes the triage accuracy

from station j to station i, given station j is a triage station. η_{ji} =1 means perfect inspection and 0 otherwise.

3.3.3 Simulation Model for Disease Propagation

To facilitate disease propagation analysis and its impact on dynamic resource allocation, an efficient simulator/optimizer decision support system is critical as it enables scenario-based analysis. Different types of simulation model have been proposed [14][15][67][81] for analyzing the disease spread across one or several regions. Our work is the first to focus on the interactions of dispensing operations and disease propagation inside the mass-dispensing clinic, and provide insight on designing the mitigation strategies [55].

We take advantage of the real-time simulation-optimization environment that CDC-RealOpt-POD offers and incorporates our disease propagation module within its environment. It allows incorporation of system stochasticity and offers insights into the intra-POD disease transmission. Furthermore, simulation-based disease propagation module also enables broad range sensitivity and scenario-based analysis and allows epidemiologists and POD logistics experts test alternative POD process layout and analyze the tradeoffs between infection risks and operational efficiencies. Figure 24 shows how the disease propagation module is incorporated into CDC-RealOpt-POD.

The disease propagation module incorporates the population size-dependent contact-based infection force to provide accurate statistics estimate under general POD designs. More specifically, traditional infection rate-based disease propagation model does not appear to capture the phenomenon that larger batch size may lead to more infections in the batch process station. CDC-RealOpt-POD can keep track of each single contact during the simulation process and thus avoid the difficulty of ODE systems (3.3.2.4) in

estimating intra-POD contact number, which usually varies among different stations and is highly influenced by the service type, i.e. individual service versus group service.

However, these advantages come with a price. First of all, simulation experiments must be carefully designed in order to draw useful conclusions, as the outcome is only as good as the input parameters provided. Second, regardless of the number of replicates, the results can only serve as estimates of disease propagation risk. Further, simulation-based disease propagation module is computational intensive. While ODE models can be scaled up readily as population size increases, scalability in simulation-based disease propagation module can be highly limited. CDC-RealOpt-POD offers a rapid simulation/optimization computational environment. Figure 24 shows the modeling and computational schema of integrating simulation-based disease propagation into the CDC-RealOpt-POD environment.

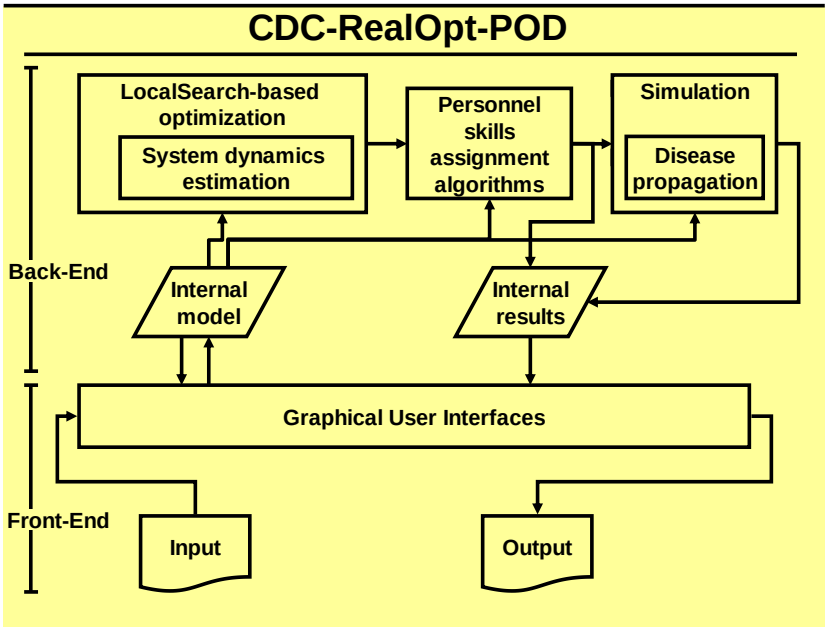


Figure 24 Integration of disease propagation within the CDC-RealOpt-POD environment

3.3.4 Generalized Compartmental Model for Disease Propagation

By examining the nature of a compartmental model, we notice that the compartmental model can in fact be described in a way where the model structure and the model configuration are separated. Model structure refers to the *abstract* form of network representation of the compartmental model, and model configuration means the *real* precedence relations between different disease stages. This modeling scheme not only enhances the modeling capability, it also makes the modular implementation of such model possible.

We introduce the modular implementation and incorporated it within the CDC-RealOpt-POD environment. The implementation entails a graphical user-input construction that allows great flexibility for users to examine their own epidemiology models.

We describe the mathematical modeling abstraction scheme in detail, and use the 6-stages model for illustration.

3.3.4.1 Mathematical Formulation

We first define the stages for our general disease propagation model.

- All Stages: This set contains all the stages that are required in a compartmental model. These stages can be related to the disease progression itself, such as susceptible or infectious, or related to the interaction of disease progression and POD operations if any, such as vaccination. Let $\mathbf{\Phi}$ denote all stages. In 6-stages model, $\mathbf{\Phi} = \{S, E, P, A, I, R, D, V\}$.

- Passive/Active Stages: Passive stages are those stages under which an entity will not change its status spontaneously; Active stages are those stages under which an

entity will change its status spontaneously. Let $\mathbf{\Phi}_{\mathrm{P}}$ denote the passive stages, and $\mathbf{\Phi}_{\mathrm{A}}$ denote the active stages. In 6-stages model, $\mathbf{\Phi}_{\mathrm{P}} = \{S, R, D, V\}$, $\mathbf{\Phi}_{\mathrm{A}} = \{E, P, A, I\}$.

- Vulnerable/Contagious Stages: Vulnerable stages are those stages under which an entity can still be infected; Contagious stages are those stages under which an entity can infect other entities who are vulnerable. Let $\mathbf{\Phi}_{\mathrm{V}}$ denote the vulnerable stages, and $\mathbf{\Phi}_{\mathrm{C}}$ denote the contagious stages. In 6-stages model, $\mathbf{\Phi}_{\mathrm{C}} = \{P, A, I\}$, $\mathbf{\Phi}_{\mathrm{V}}=\{\mathrm{S}\}$.

- Living/Non-living Stages: Living stages are those stages under which an entity will stay in the model; Non-living stages are those stages under which an entity will be removed from the model. Let $\mathbf{\Phi}_{\mathrm{L}}$ denote the living stages, and $\mathbf{\Phi}_{\mathrm{N}}$ denote the non-living stages. In 6-stages model, $\mathbf{\Phi}_{\mathrm{L}} = \{S, E, P, A, I, R, V\}$, $\mathbf{\Phi}_{\mathrm{N}} = \{D\}$.

To model the POD operations and disease spread under this separation schema of model structure and model configuration, we use the following notations:

We assume that there are K stations in the POD, Let N_i denote the number of living individuals at station i, $i\epsilon\{1, 2, \cdots, \mathrm{K}\}$, and $N_i = \sum_{\vartheta\in\mathbf{\Phi}_{\mathrm{L}}} \vartheta_i$. When i=0, it denotes the number of living individuals who have not yet entered POD. We have λ_i denote the service rate at station i, $i\epsilon\{1, 2, \cdots, \mathrm{K}\}$. ψ_i is a binary indicator where ψ_i=1 if station i is the very first station (POD entrance) of POD, and 0 otherwise, for $i\epsilon\{1, 2, \cdots, \mathrm{K}\}$. This indicator is used to connect the outer-POD formulation and intra-POD formulation. $\mathbf{q}_i$ is the vector in which each element q_{ji} means the transition probability from station j to station i in POD operations. $q_{ji} = 0$ if station j is not an immediate predecessor of station i. Lastly, r_a denote the arrival rate of POD. It can be extended to $r_a(t)$ if the arrival rate is heterogeneous with time.

We define $\eta_{\vartheta i}$ as a continuous number bounded in the range $0 \leq \eta_{\vartheta i} \leq 1$ for individual at station *i* at stage ϑ. $\eta_{\vartheta i}$ is used to model the fraction of outflow of people at stage ϑ from station *i*. When $\eta_{\vartheta i}$=1, it means all individuals at stage ϑ, after receiving service at station *i*, will proceed to successive station(s). When $\eta_{\vartheta i}$=0, it means all individuals at stage ϑ, after receiving service at station *i*, will be removed from regular POD operations for special treatment or be removed for other purposes in general. This factor $\eta_{\vartheta i}$ is generic and convenient in modeling situations such as people at Symptomatic stage will be removed from POD after Triage station, or people at Susceptible stage will be removed from POD after Vaccination station. In the latter case, people at Susceptible stage are only "conceptually" removed and they will be added to the successive station as people at Vaccinated stage.

We extend the original definition of function $h_i(\vartheta)$ and re-define $h_i(\vartheta) = \lambda_i \cdot min\{n_i, N_i\} \cdot \frac{\vartheta_i}{N_i} \cdot (1 - \eta_{\vartheta i})$, $i\epsilon\{1, 2, \cdots, \mathrm{K}\}$ and $\vartheta \in \mathbf{\Phi}_{\mathrm{L}}$. For completeness, we define $h_i(\vartheta) \equiv 0$ for $\vartheta \in \mathbf{\Phi}_{\mathrm{N}}$. Furthermore, we have service rate vector $h_i(\mathbf{\Phi}) \equiv \langle h_i(\vartheta) \rangle|_{\vartheta \epsilon \mathbf{\Phi}}$, and service rate matrix $\mathbf{H}(\mathbf{\Phi}) \equiv [h_1(\mathbf{\Phi})^{\mathrm{T}}, h_2(\mathbf{\Phi})^{\mathrm{T}}, \cdots, h_{\mathrm{K}}(\mathbf{\Phi})^{\mathrm{T}}]$.

We use $\boldsymbol{\mu}_{\mathbf{\Phi}} \equiv \langle \mu_\vartheta \rangle|_{\vartheta \epsilon \mathbf{\Phi}}$ as a vector form of the transition rate of all the stages in the model. One can observe that only the active stages have well-defined transition rates. For passive stages, we define $\mu_\vartheta = 0$ for $\vartheta \in \mathbf{\Phi}_{\mathrm{P}}$ for completeness. Identical to the definition in section 3.4.2, γ denotes the contact number, and δ denotes the contact coefficient.

Let matrix $\mathbf{C}$ denote the disease contagious matrix. $\mathbf{C} \equiv \langle c_{ab} \rangle|_{a \epsilon \mathbf{\Phi}, b \epsilon \mathbf{\Phi}}$, where c_{ab}=-1 if $a\epsilon\mathbf{\Phi}_{\mathrm{V}}$ and b$\epsilon\mathbf{\Phi}_{\mathrm{C}}$; $c_{a_{succ}b}$=+1, where a_{succ} means direct successive stage of stage *a*.

Let matrix $\mathbf{D}$ denote the disease development matrix. $\mathbf{D} \equiv \langle d_{ab} \rangle|_{a \epsilon \mathbf{\Phi}, b \epsilon \mathbf{\Phi}}$, where d_{ab} means the transition probability from stage *b* to stage *a*.

Let matrix $\mathbf{J}_i$ denote the stage-jump matrix for station *i*. $\mathbf{J}_i \equiv \langle u_{ab} \rangle|_{a\epsilon\mathbf{\Phi}, b\epsilon\mathbf{\Phi}}$, where u_{ab}=1 if the process at station *i* can transform individual from stage *b* to stage *a*. u_{ab}=0 otherwise. For example, at Vaccination station, patients can be transformed from Susceptible stage to Vaccinated stage.

We define the state vector for outer-POD individuals in the ODE formulation as $\mathbf{y}_0 \equiv \langle \vartheta_0(t) \rangle|_{\vartheta\epsilon\mathbf{\Phi}} \equiv \langle \vartheta_0 \rangle|_{\vartheta\epsilon\mathbf{\Phi}}$, and first-order derivative in terms of time *t* is $\mathbf{y}_0' \equiv \langle \frac{d}{dt}\vartheta_0(t) \rangle\Big|_{\vartheta\epsilon\mathbf{\Phi}} \equiv \langle \vartheta_0' \rangle|_{\vartheta\epsilon\mathbf{\Phi}}$. Define the state vector for intra-POD individuals at station *i* in the ODE formulation as $\mathbf{y}_i \equiv \langle \vartheta_i(t) \rangle|_{\vartheta\epsilon\mathbf{\Phi}} \equiv \langle \vartheta_i \rangle|_{\vartheta\epsilon\mathbf{\Phi}}$, and first-order derivative in terms of time *t* is $\mathbf{y}_i' \equiv \langle \frac{d}{dt}\vartheta_i(t) \rangle\Big|_{\vartheta\epsilon\mathbf{\Phi}} \equiv \langle \vartheta_i' \rangle|_{\vartheta\epsilon\mathbf{\Phi}}$.

Now we can reformulate the compartmental model for outer-POD disease propagation Eq (3-1) to Eq (3-7) into the following Eq (3-16).

$$\mathbf{y}_0' = -\frac{r_a}{N_0} \cdot \mathbf{I} \cdot \mathbf{y}_0 + \frac{min\{N_0, \gamma\}}{N_0} \cdot \delta \cdot \mathbf{C} \cdot \mathbf{y}_0 + [\mathbf{D} - \mathbf{I}] \cdot \mathrm{diag}(\boldsymbol{\mu}_{\mathbf{\Phi}}) \cdot \mathbf{y}_0 \qquad (3\text{-}16)$$

where:

- $r_a/N_0 \cdot \mathbf{I} \cdot \mathbf{y}_0$ denote the POD arrival rate,
- $min\{N_0, \gamma\}/N_0 \cdot \delta \cdot \mathbf{C} \cdot \mathbf{y}_0$ represent the rate of disease infection, and
- $[\mathbf{D} - \mathbf{I}] \cdot \mathrm{diag}(\boldsymbol{\mu}_{\mathbf{\Phi}}) \cdot \mathbf{y}_0$ denote the rate of disease progression.

To reformulate the compartmental model for intra-POD disease propagation Eq (3-8) to Eq (3-15) into the following Eq (3-17), we have:

$$\mathbf{y}_i' = \psi_i \cdot \frac{r_a}{N_0} \cdot \mathbf{I} \cdot \mathbf{y}_0 + \delta \cdot \mathbf{C} \cdot \mathbf{y}_i + [\mathbf{D} - \mathbf{I}] \cdot \mathrm{diag}(\boldsymbol{\mu}_{\boldsymbol{\Phi}}) \cdot \mathbf{y}_i - \mathrm{diag}\big(h_i(\boldsymbol{\Phi})\big) \cdot \mathbf{y}_i + \mathbf{H}(\boldsymbol{\Phi}) \cdot \mathbf{q}_i + \mathbf{J}_i \cdot \mathbf{H}(\boldsymbol{\Phi}) \cdot \mathbf{q}_i \quad (3\text{-}17)$$

In Eq (3-17):

- $\psi_i \cdot r_a/N_0 \cdot \mathbf{I} \cdot \mathbf{y}_0$ denote POD arrival rate,
- $\delta \cdot \mathbf{C} \cdot \mathbf{y}_i$ represent rates of disease infection,
- $[\mathbf{D} - \mathbf{I}] \cdot \mathrm{diag}(\boldsymbol{\mu}_{\boldsymbol{\Phi}}) \cdot \mathbf{y}_i$ denote rate of disease progression,
- $\mathrm{diag}\big(h_i(\boldsymbol{\Phi})\big) \cdot \mathbf{y}_i$ denote outflow rate of POD operation,
- $\mathbf{H}(\boldsymbol{\Phi}) \cdot \mathbf{q}_i$ denote stage-independent inflow rate of POD operation, and
- $\mathbf{J}_i \cdot \mathbf{H}(\boldsymbol{\Phi}) \cdot \mathbf{q}_i$ denote stage-dependent inflow rate of POD operation.

3.3.4.2 Implementation of a Unified Platform for POD Operations and Disease Compartmental Model

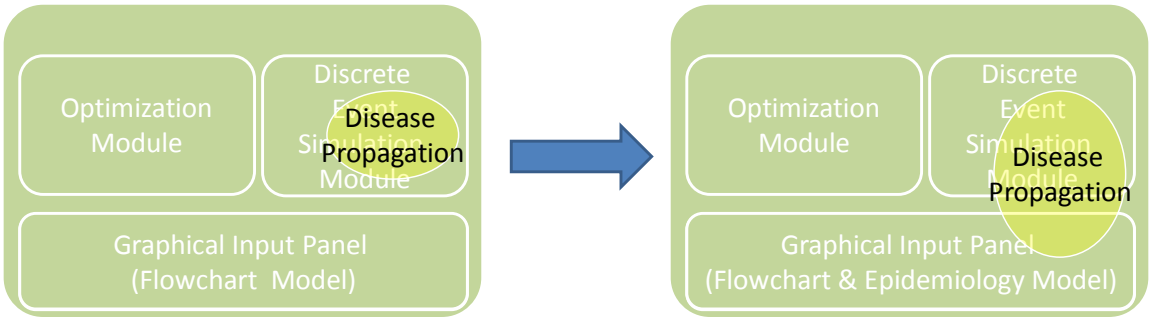
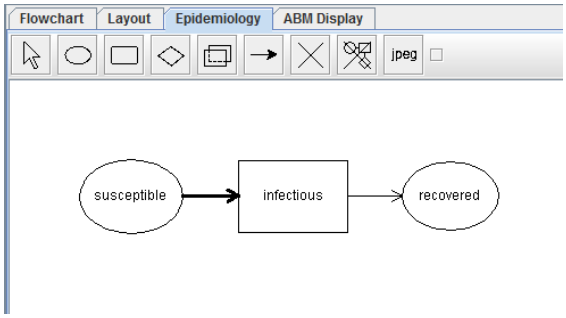
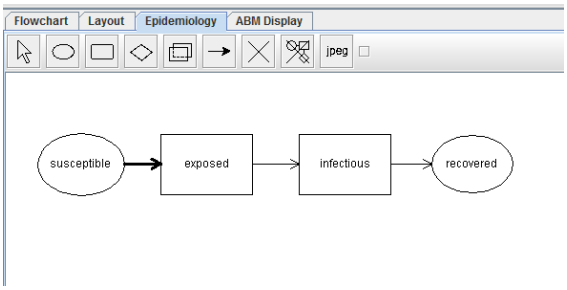


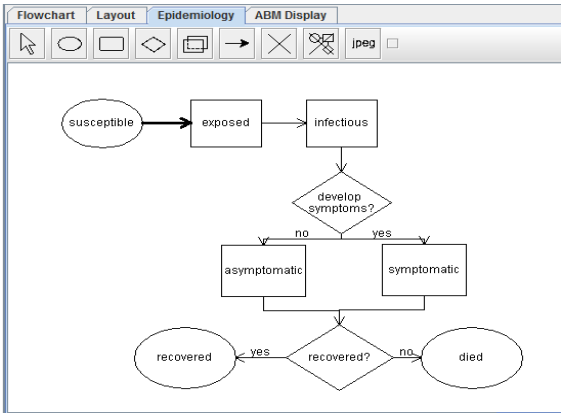
Figure 25 System extension from 6-stages compartmental model to general compartmental model



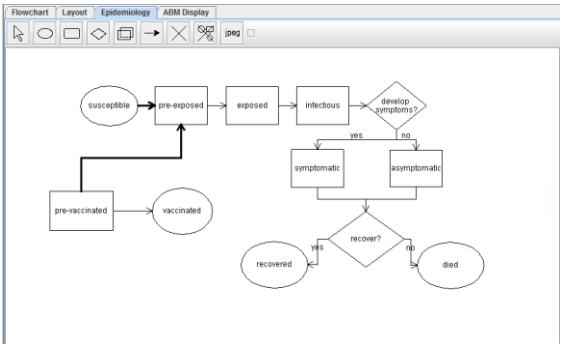
(a) SIR Model



(b) SEIR Model



(c) SEPAIR (6-stages) Model



(d) Extended SEPAIR (6-stages) Model

Figure 26 Illustrative user-defined compartmental models

3.3.5 Numerical Validation

In this section we show the numerical results of disease propagation in a flu-vaccination clinic. We used 36-hour period as the time horizon for both analytical and simulation computations.

Section 3.3.5.1 first describes the parameter values for both POD operations and disease propagation. Section 3.3.5.2 presents the analytic ODE solutions and compares them to the CDC-RealOpt-POD simulation results. We highlight the importance of contact based infection force for disease propagation analysis in dispensing clinics. Section 3.3.5.3 shows the base case results and provides fundamental understanding

about the magnitude and dynamics of intra-facility disease propagation. In section 3.3.5.4 we provide sensitivity analysis to explore possible strategies to mitigate disease propagation. Significance of mitigation strategies under different disease infectivity is discussed.

3.3.5.1 Parameter Values

Figure 27 shows the floor plan for a flu vaccination clinic, and Table 4 summarizes the service times at each station. The throughput is set to 1000 individuals per hour, with 36000 individuals being served over a 36-hour horizon. As a base case study, we assume that triage is 100% accurate. Thus every patient with symptoms will be detected and sent to hospital or other medical clinic, and that there is no false-negative. Next, we will analyze triage sensitivity to examine its influence on mitigating the intra-POD disease propagation.

Details of patients flow inside POD are as follow: Patients arrive in the POD individually and are then accumulated in the batching area (Batch before orientation) to form a group with predetermined size of 40. The patient inter-arrival times is assumed to follow the exponential distribution, where the arrival rate is 16.67 per minute (given hourly throughput is 1000) and stays consistent over the 36-hour dispensing period. After orientation station patients are again served individually. 50% of patients are assumed to require form filling assistance, while the rest fill out the forms by themselves. In addition, 50% of patients are assumed to require further medical screening, while the rest can go to vaccination station directly. Among those patients who need further medical screening, 10% are assumed to have feasible alternate medication and are sent to other medical facilities, the rest will proceed to the vaccination station**.**

We note that the orientation station and corresponding "batch before orientation" and "breakup batches after orientation" will not be used for validation when comparing simulation and ODE performance, since the ODE model is not capable of modeling the flow accumulation in the batch area.

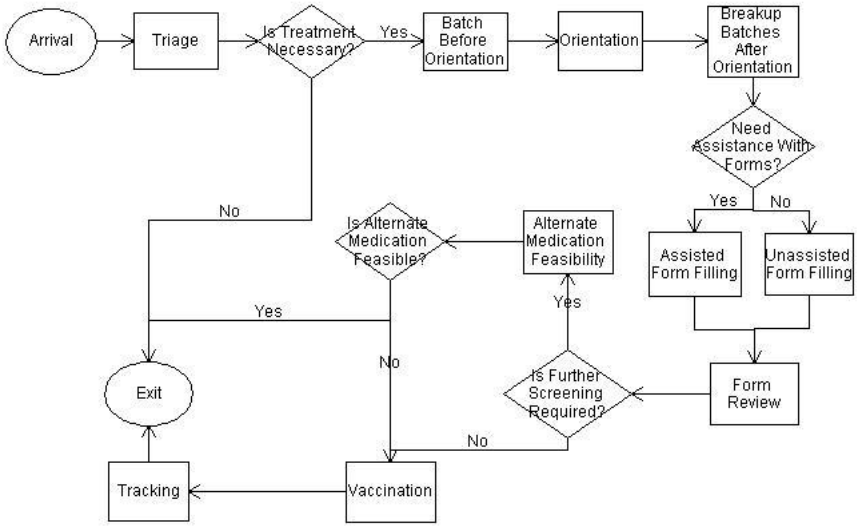


Figure 27 Floorplan for flu vaccination

Table 4 Process service times in this floorplan

Station Name	**Process Time**	**Unit**
Triage	Tria(10, 30, 120)	Second
Orientation	Tria(480, 600, 900)	Second
Assisted Form Filling	Tria(10, 60, 120)	Second
Unassisted Form Filling	Tria(10, 60, 120)	Second
Form Review	Tria(10, 30, 90)	Second
Alternate Medication Feasibility	Tria(10, 30, 120)	Second
Vaccination	Tria(50, 120, 300)	Second
Tracking	Tria(10, 30, 60)	Second

Table 5 lists the parameters for disease progression and transmission. These parameters are consistent with recent work on pandemic flu containment and mitigation

strategies [23][24][29][68][74]. The 6-stages disease propagation model is employed in the analysis. Mean exposed dwell time means the average time for patient in exposed stage progressing to infectious stage. Mean infectious, asymptomatic, and symptomatic dwell times are defined in analogous way. Initial percentages are used to model the size of initial disease outbreak. In the base case it is assumed that 5% of the entire population is infectious and the remaining are healthy but susceptible. The symptomatic proportion is assumed to be 67%, which means on 67% of infectious patients the symptoms will develop. Lastly, while the basic reproduction number (BRN, R_0) and infection rate are derived directly from the literature, we estimated the average contact number and contact coefficient by using the known R_0 and infection rate.

Unlike the POD operation parameters where we can perform time-motion study to capture their stochastic service distributions, parameters for the disease propagation are usually more difficult to estimate accurately. For example, the estimate of contact number or contact coefficient is difficult to obtain if the flu strain is new. The initial infectious percentage can only be roughly estimated, medication dispensing may be launched before complete information is gathered, and more importantly there does not exist a system/model in place to collect complete infectious information. In the sensitivity analysis, we vary these parameters to provide insights of dynamic and magnitudes of disease spread to compensate for the potential lack of accurate estimates. Mitigation strategies under different possible scenarios are discussed.

Table 5 Parameters for disease transmission and progression

Disease Stage	Initial Percentage(%)	Mean Dwell Time
Susceptible	95	
Exposed	0	1 day
Infectious	5	1 day
Asymptomatic	0	3 days
Symptomatic	0	3 days
Recovered	0	

BRN (R_0)	2	
Infection Rate	0.5	/day
Contact Number (γ)	193	
Contact coefficient (δ)	0.18	E-5/min

3.3.5.2 Model Validations

For benchmarking, the analytical compartmental model solutions are first compared to the simulation results using the base case. The advantages of using simulation program for disease propagation, despite its intensive computational requirement, are then addressed. The importance of contact based force of infection is highlighted by contrasting results from infection rate-based propagation and contact based propagation.

We focus on the ccumulated number of intra-POD infections from ODE model and from CDC-RealOpt-POD simulation. In CDC-RealOpt-POD simulation, we used two different types of probability distributions, triangular distribution and exponential distribution, to help demonstrate the underestimation in the ODE model. By comparing the number of intra-POD infections derived from ODE model and from the CDC-RealOpt-POD simulation, Figure 28 shows that the ODE model tends to result in lower number of intra-POD infection. The underestimation of solving the ODE model comes from the variation of processing time modeled in the open queueing network. Recall that

medical dispensing in a POD is essentially an open queueing network with practical conditions involved. It is known that exponential distribution has larger variation than the triangular distribution. ODE model is basically the mean-value analysis (MVA) and hence is not capable of capturing processing time variation. Processing time variation in ODE model is thus zero. Furthermore, from the basic principle of queueing theory [32][38] we know that larger variation in processing time contributes to longer waiting time and queue length as well. This directly explains that patients will stay longer in the POD if the variation of processing time is larger, and this incurs higher opportunity of infection.

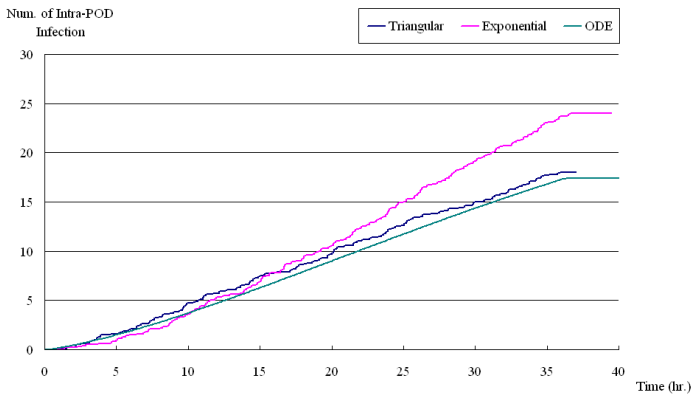


Figure 28 Underestimation of intra-POD Infections in ODE Model [56]

We note that both models use the same optimal staffing that is obtained by CDC-RealOpt-POD. ODE does not offer such ability as to optimize personnel allocation. As expected, running ODE is computationally more efficient than using CDC-RealOpt-POD simulation in estimating number of intra-POD infection. In addition, ODE has virtually no limitation on the scale of population it can handle. Determining personnel allocation in ODE may, however, require integration between ODE model and other optimization tools.

It is intuitive that larger batch size will contribute to higher number of infection, this is confirmed in Figure 29 when we perform the disease spread via the simulation framework. Figure 29 shows that infection-rate based propagation does not capture the phenomenon that larger batch size can lead to more infection inside the orientation station. Rather, the increase of number of intra-POD infection comes from the increase in the batching area (batch before orientation). In this area, patients need to wait to form a batch before going to the orientation station. Larger batch size immediately means, given identical arrival rate, longer time waiting in the batching area to form a batch, and hence higher number of infection in the batching area.

By implementing the force of infection to contact based propagation, Figure 30 shows that the number of infection in orientation station increases as batch size increases. As for batch before orientation, Figure 30 shows a higher rate of increase compared to that of Figure 29. This means that the increase of infection in batching area in Figure 30 results from both the longer waiting time and the higher number of contacts.

Figure 29 and Figure 30 also reveal that the infection rate-based propagation model tends to underestimate the number of infection when batch size is large. The underestimation comes from the fact that, in infection rate-based model, the force of infection is defined as the number of people each infectious (and (a)symptomatic) individual can infect per unit time, larger batch size actually amortizes the probability of getting infected for each susceptible individual. The overestimation under smaller batches can also be explained in similar but reversed way.

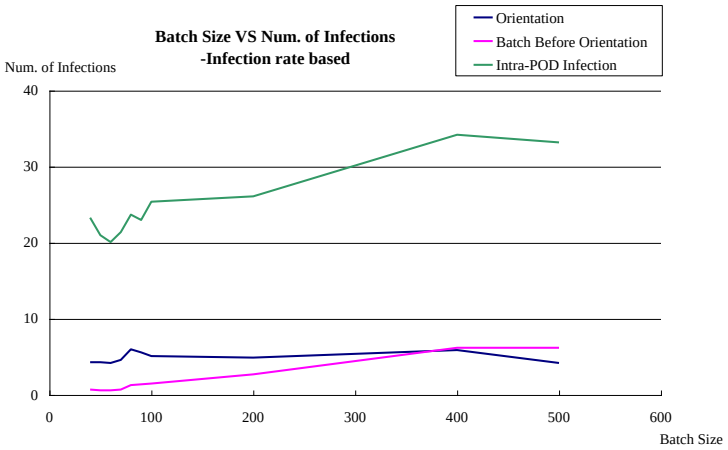


Figure 29 Number of infections (infection rate-based) under different batch sizes

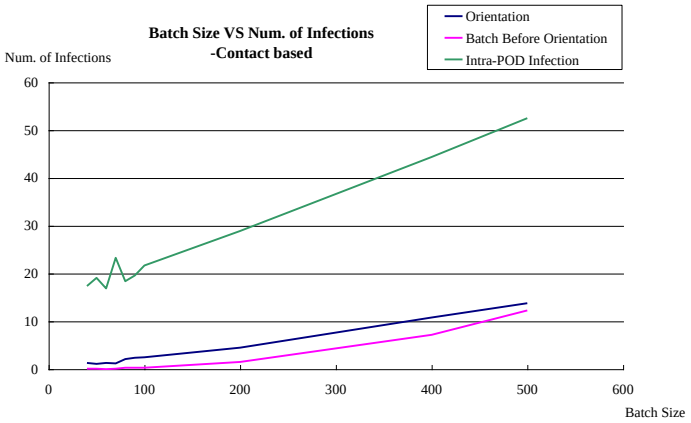


Figure 30 Number of infections (contact based) under different batch sizes

The analysis in 3.3.5.3 and 3.3.5.4 are performed using the CDC-RealOpt-POD simulation-optimization environment.

3.3.5.3 Base Case Results

Figure 31 and Figure 32 depict the dynamics of number of intra-POD individuals under different disease stages and intra-POD disease progression, respectively. Figure 31 provides the primary information that are present in most epidemiology studies, Figure

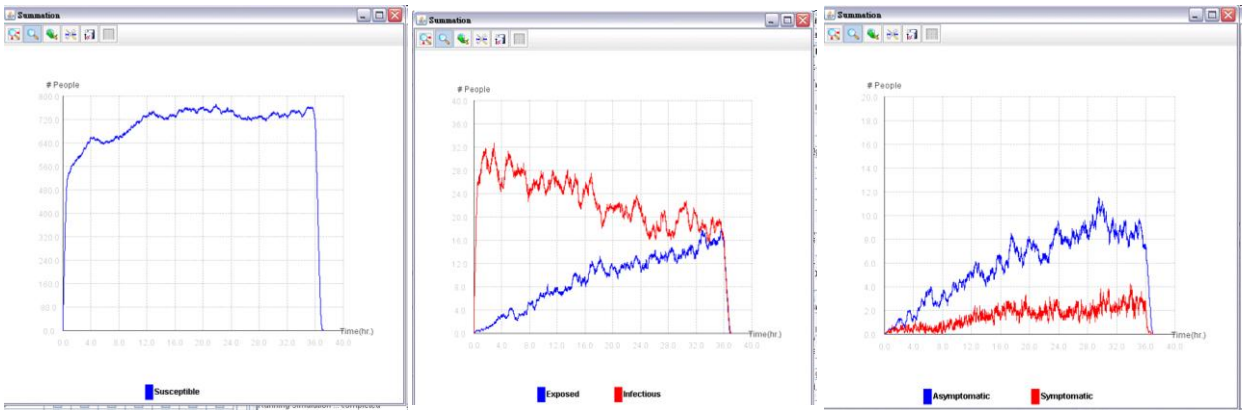
32 helps the epidemiologists and POD logistic experts in examining alternative POD designs and analyzing the tradeoffs between operational efficiency and disease spreading.

Figure 31(a)(b)(c) together depict that, the number of people inside the POD (both being served and waiting) stay at a consistent level throughout. In Figure 31(b) we observed that the number of exposed individuals increases from 0 to 16 after 36-hour operations, but the number of infectious individuals decreases by almost the same amount. Figure 31(c) validates the effect of implementing triage operation in the POD entrance. Given 67% of symptomatic proportion, the number of symptomatic individuals only increases from 0 to 3, which is significantly lower when compared to 9 in asymptomatic individuals.

Figure 32(a) informs the POD planners and epidemiologists that implementing "perfect" mitigation strategies, if possible, can decrease the intra-POD infection by at most 18 in the given scenario. Notice that "perfect" mitigation strategies, if exist, need to improve both triage accuracy to 1 and decrease contact coefficient inside POD to 0, since the number of individuals with symptoms developed after entering the POD cannot be reduced by improving triage accuracy. Figure 32(b) shows that 18 infectious individuals develop symptoms inside the POD in the given scenario. Figure 32(a) shows that 18 people are infected, meaning they move from susceptible stage to exposed stage, because there is false negative in Triage station, and also disease is still possible to spread inside POD because of non-zero contact coefficient. In fact, as long as contact coefficient can be reduced to zero, no infection will occur even if there are symptomatic individuals inside. However, since this is hard to achieve, it is better to reduce both of them to decrease intra-POD infection.

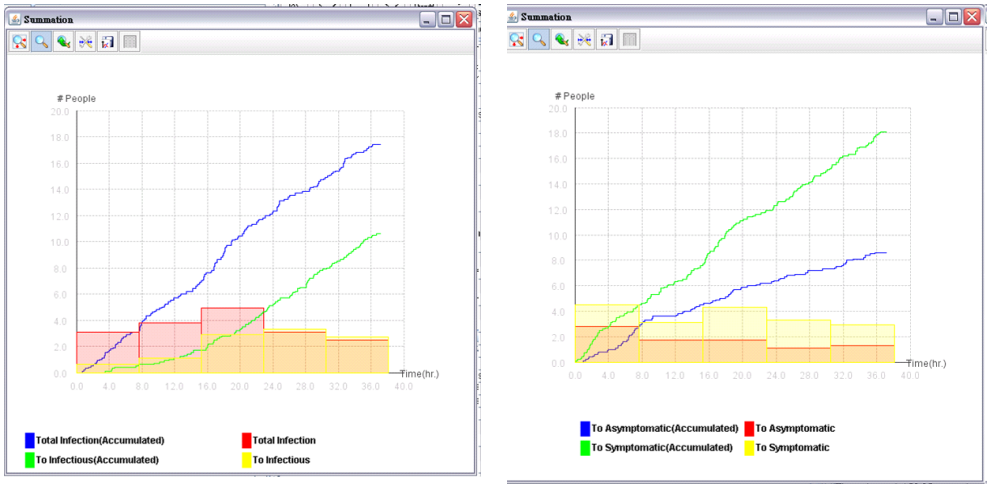
Computational time of the base case scenario is 61.97 CPU seconds. The computation time includes optimizing personnel allocation and 10 replicates of simulation for

capturing system dynamics. Disease propagation module is embedded in the simulation and hence is run 10 times as well.



(a) (b) (c)

Figure 31 The dynamics of number of intra-POD individuals. (a) shows the number of susceptible individuals varies over the 36-hour period. (b) shows number of exposed and infectious individuals, and (c) shows number of asymptomatic and symptomatic individuals.



(a) (b)

Figure 32 The intra-POD disease progression. (a) cumulative number of intra-POD infections and cumulative number of disease progressions from exposed stage to infectious stage. (b) cumulative number of disease progression from infectious stage to asymptomatic and symptomatic stage.

3.3.5.4 Strategies for Mitigating intra-POD Disease Propagation

Strategies, such as isolating, contact tracing, and quarantining symptomatic individuals for mitigating and containing disease outbreak across a region have been studied extensively [23][24][26][89]. In this section, we focus on exploring three possible intra-POD mitigation strategies and evaluate their significance under different disease infectivity. These mitigation strategies are i) triage accuracy, ii) POD throughput, and iii) batch size (if batch process is included in the POD operation). We used symptomatic proportion and contact coefficient to define disease infectivity. We also explore how the size of initial outbreak will influence the disease propagation under different mitigation strategies.

As sensitivity analysis, each parameter is set within a range so that the trend of number of intra-POD infection can be revealed. The initial infectious percentage rages from 1% to 10%, and is incremented by 1%. It represents the initial percentage of infectious individuals among the entire population. Symptomatic proportion ranges from 10% to 100%, and incremented by 10%. 100% symptomatic proportion means all infectious individuals will develop symptoms. 10% symptomatic proportion means on average only 10% of infectious individuals will develop symptoms, and the remaining 90% will not. Lastly, contact coefficient ranges from 0.1E-5/min to 1.0E-5/min and is incremented by 0.1E-5.

3.3.5.4.1 Triage Accuracy versus intra-POD Disease Propagation

This section quantifies and evaluates the effect of improving triage accuracy under different size of initial outbreaks, different symptomatic proportions, and different contact coefficient. The number of intra-POD infections is used as the response value. To understand its impact, triage accuracy is set to a range from 0.1 to 1 and is incremented by 0.1. Triage accuracy =1 means all the symptomatic individuals are detected, and triage accuracy=0.1 means on average 1 out of 10 symptomatic individuals is detected.

Figure 33 shows the number of intra-POD infections under different triage accuracy and initial infectious percentage. Here, the 6-stage epidemiology model is used. Throughput=36000 over a 36-hour period. Symptomatic proportion is 67%, contact number is 193 (for outer-POD disease propagation), and contact coefficient =0.18E-5/min. Initial percentage for susceptible depends on the initial infectious percentage, and we assume initial exposed, asymptomatic, and symptomatic percentages are 0. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 55.52 to 114.6 CPU seconds.

Figure 33 shows that, by improving the triage accuracy from 0.1 to 1, the number of infections can be reduced by 35% when initial infectious percentage is 1%, and 64% when initial infectious percentage is 10%. This confirms the significance of triage operation while confronting larger scope of disease outbreak or bio-attack. Our simulation also shows that the trend is identical when hourly throughput is equal to 500, 1000, 1500, and 2000. This further shows the necessity of high triage accuracy in both centralization and decentralization scenarios.

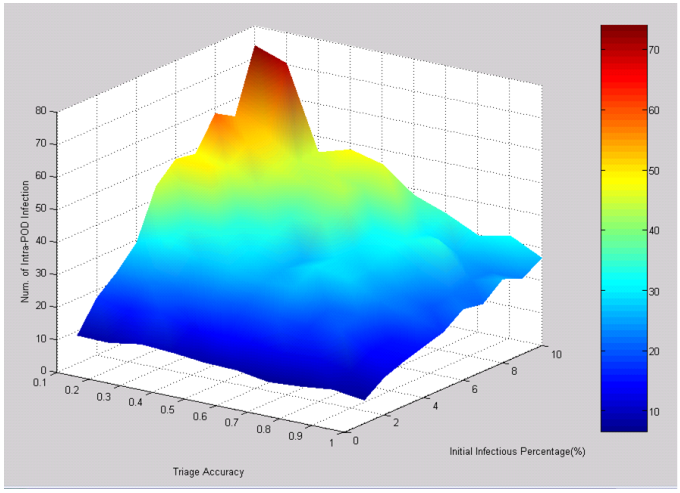


Figure 33 Number of intra-POD infections under different triage accuracy and initial infectious percentage

Figure 34 shows the number of intra-POD infection under different triage accuracy and symptomatic proportion. Here, the 6-stage epidemiology model is used. Throughput=36000 over a 36-hour period. Contact number is 193 (for outer-POD disease propagation), and contact coefficient =0.18E-5/min. Initial percentage for susceptible is 95%, for infectious is 5%. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 57.13 to 83.99 CPU seconds.

Figure 34 shows that, by improving the triage accuracy from 0.1 to 1, the number of infections can be reduced by 16% when symptomatic proportion is 10%, and 66% when symptomatic proportion is 100%. It is not surprising that triage station does not work well if most of the infectious individuals do not develop symptoms. The sensitivity analysis on different triage accuracies and symptomatic proportions emphasizes the importance of using the SEPAIR 6 stages propagation model instead of the SEIR 4 stages model. SEIR model is not conducive for examining the effect of improving triage accuracy. Although it is well-known that improving triage accuracy is more important when disease has higher probability of explicitly showing symptoms, using SEPAIR 6 stages propagation model helps quantify the trade-off and thus facilitates POD layout design, in particular the inclusion of the triage station, given that the infected clients have sufficient explicit disease characteristics.

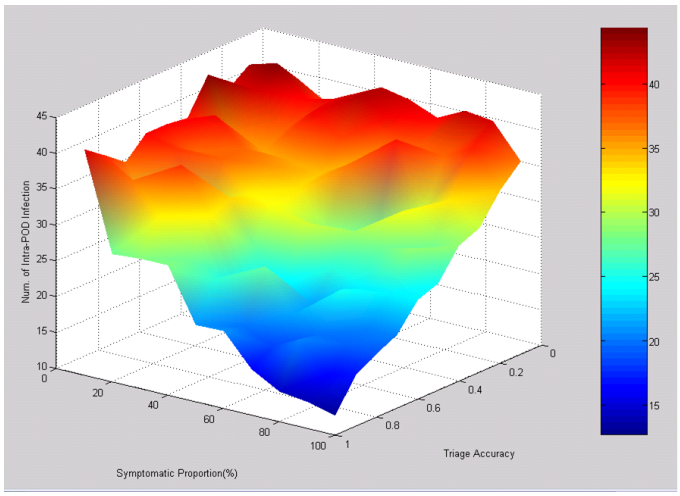


Figure 34 Number of intra-POD infections under different triage accuracy and symptomatic proportion

Figure 35 shows the number of intra-POD infection under different triage accuracy and contact coefficient (delta). Here, the 6-stage epidemiology model is used. Throughput=36000 over a 36-hour period. Symptomatic proportion is 67%, and contact number is 193 (for outer-POD disease propagation). Initial percentage for susceptible is 95%, for infectious is 5%. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 63.44 to 110.93 CPU seconds.

Figure 35 shows that by improving the triage accuracy from 0.1 to 1, the number of infections can be reduced by 55% when contact coefficient is 0.1E-5/min, and 60% when contact coefficient is 1.0E-5/min.

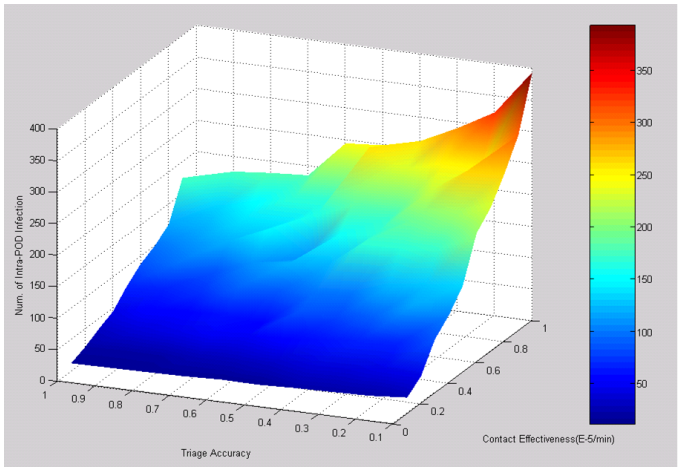


Figure 35 Number of intra-POD infections under different triage accuracy and contact coefficient

3.3.5.4.2 Centralization versus Decentralization Dispensing

Considering the non-homogeneous population density, it has been proved that heterogeneous mode of dispensing is preferable. Lee et al. [64] showed that for urban areas such as New York cities with high population densities, the most cost-effective dispensing strategy is to use “mega PODs” with large facilities for housing dispensing operations. In contrast, suburban locals can be well served by utilizing high schools, churches, or recreation centers for POD sites. Different modes of dispensing correspond to different levels of throughput requirement. By centralizing the dispensing operations it means we tend to have PODs with high capacity and hence this leads to smaller total number of PODs required. Decentralization employs more but small-capacity POD sites running with lower staff and lower throughput. In this section we investigate the trade-offs between centralized/decentralized dispensing schemes and intra-POD infection under different disease characteristics. We briefly summarize results for which throughputs are set in four different levels: 500/hour, 1000/hour, 1500/hour, and 2000/hour.

Figure 36 shows the number of intra-POD infection under different 36-hour period throughput and initial infectious percentage. Here, the 6-stage epidemiology model is used. Triage accuracy=1, symptomatic proportion is 67%, contact number is 193 (for outer-POD disease propagation), and contact coefficient =0.18E-5/min. Initial percentage for susceptible depends on the initial infectious percentage, and we assume initial exposed, asymptomatic, and symptomatic percentages are all 0. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 38.52 to 134.37 CPU seconds.

Figure 36 shows that reduction of intra-POD infection by decentralization (from 2000/hr to 500/hr) increases from 37% to 59% when initial infectious percentage increases from 1% to 10%. This implies that decentralization is a better strategy for mitigating disease spread.

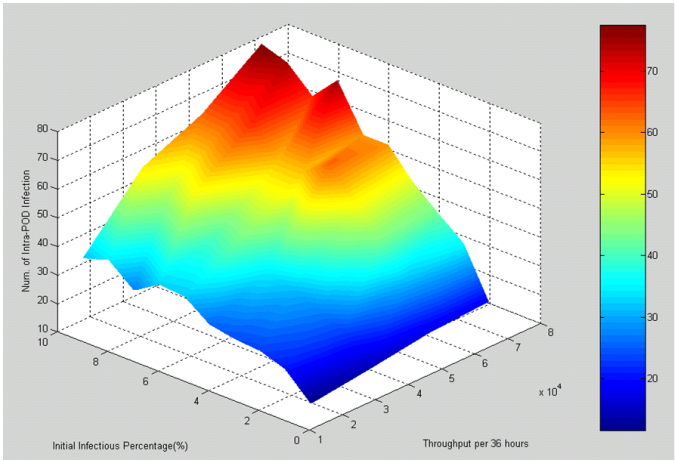


Figure 36 Number of intra-POD infections under different 36-hour period throughput and initial infectious percentage

Figure 37 shows the number of intra-POD infection under different 36-hour period throughput and symptomatic proportion. Here, the 6-stage epidemiology model is used. Triage accuracy=1, contact number is 193 (for outer-POD disease propagation), and contact coefficient =0.18E-5/min. Initial percentage for susceptible is 95% and for

infectious is 5%. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 37.5 to 110.57 CPU seconds.

In Figure 37, the average reduction of intra-POD infection by decentralization (from 2000/hr to 500/hr) is 54% over all levels of symptomatic proportion. This implies that decentralization is generally a better strategy for mitigating disease spread. However, it is worth noticing that Figure 37 shows the average reduction from 2000 hourly throughput to 1500 is marginal (2%). This reveals that reduction of POD capacity by 25% may not be worthwhile, since the marginal gain may not compensate for potential higher operation costs for running more PODs. Lee et al [64] showed otherwise, that it is possible to run smaller PODs with non-increasing overall operation costs. From the graph, the threshold for significant gain appears when throughput is decreased from 1500 to 1000.

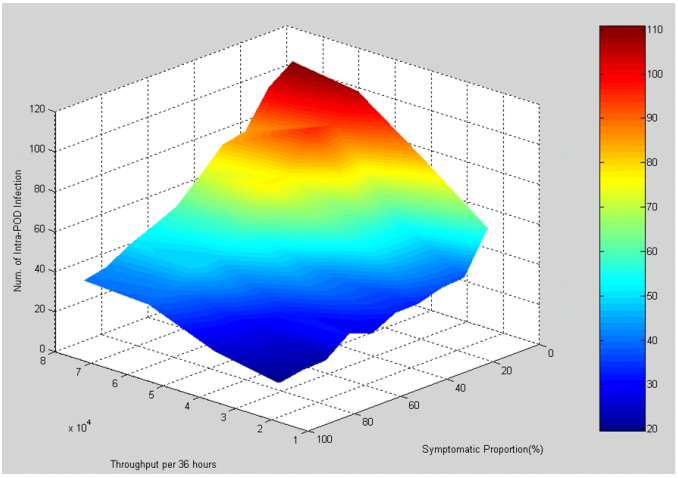


Figure 37 Number of intra-POD infections under different 36-hour period throughput and symptomatic proportion

Figure 38 shows the number of intra-POD infection under different 36-hour period throughput and contact coefficient. Here, the 6-stage epidemiology model is used. Triage

accuracy=1, symptomatic proportion=67%, and contact number is 193 (for outer-POD disease propagation). Initial percentage for susceptible is 95% and for infectious is 5%. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 42.15 to 137.29 CPU seconds.

Figure 38 shows that the average reduction of intra-POD infections by decentralization (from 2000/hr to500/hr) is, on average, 55% over all contact coefficients, indicating that decentralized dispensing is a better strategy. However, we observed similar phenomenon that reducing from 2000 hourly throughput to 1500 only contributes to 2% reduction of intra-POD infection. We expect this POD throughput value to be lower if the initial infection percentage is higher and/or the symptomatic proportion is lower.

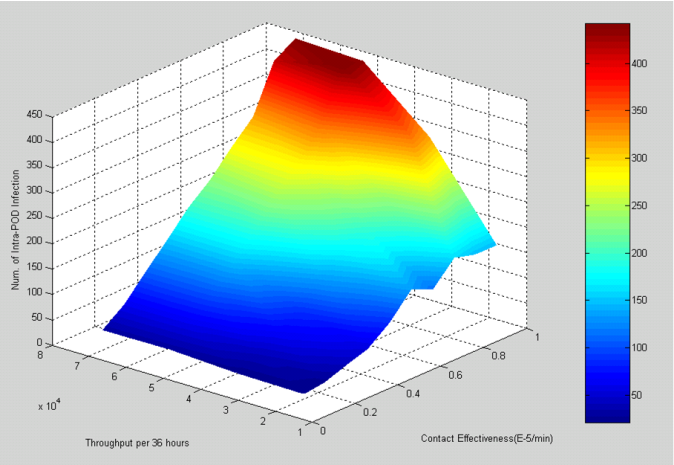


Figure 38 Number of intra-POD infections under different 36-hour period throughput and contact coefficient

3.3.5.4.3 Alternating Batch Size versus intra-POD Disease Propagation

Recall that the services or treatments in the batch process station are provided to a group with group size predetermined. The batch sizes are subject to different kinds of physical limitations. For example, the batch size for bus transportation is constrained by

the bus capacity, and the batch size for orientation is constrained by the room capacity and the function of video/audio devices. In any case, utilizing batch process in POD may be necessary in some cases, and may possibly improve throughput and decrease staff requirement. However, the benefits in operational efficiency can be offset by the higher risk of disease transmission as more people are clustered together. In this section we explore trade-offs between batch sizes and intra-POD infection under different disease characteristics. Taking physical feasibilities into account, we set batch size from 40 to 100 with 10 as increment.

Figure 39 shows the number of intra-POD infection under different batch size and initial infectious percentage. Here, the 6-stage epidemiology model is used. Throughput=36000 over a 36-hour period. Triage accuracy=1, symptomatic proportion is 67%, contact number is 193 (for outer-POD disease propagation), and contact coefficient =0.18E-5/min. Initial percentage for susceptible depends on the initial infectious percentage, and we assume initial exposed, asymptomatic, and symptomatic percentages are all 0. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 55.68 to 70.73 CPU seconds.

Figure 40 shows the number of intra-POD infection under different batch size and symptomatic proportion. Here, the 6-stage epidemiology model is used. Throughput=36000 over a 36-hour period. Triage accuracy=1, contact number is 193 (for outer-POD disease propagation), and contact coefficient =0.18E-5/min. Initial percentage for susceptible is 95%, and for infectious is 5%. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 57.34 to 81.02 CPU seconds.

Figure 41 shows the number of intra-POD infection under different batch size and contact coefficient. Here, the 6-stage epidemiology model is used. Throughput=36000

over a 36-hour period. Triage accuracy=1, symptomatic proportion-67%, and contact number is 193. Initial percentage for susceptible is 95%, and for infectious is 5%. The mean dwell time for both exposed and infectious is 1 day. For asymptomatic and symptomatic it is 3 days. Running times range from 63.59 to 71.92 CPU seconds.

Figure 39, Figure 40, and Figure 41 show the results of sensitivity analysis by modifying initial infectious percentage, symptomatic proportion, and contact coefficient, respectively. While the number of intra-POD infections responds significantly to these modified parameters as expected, all the three figures reveals that increasing batch size from 40 to 100 only lead to marginal (if any) increase in the number of intra-POD infections.

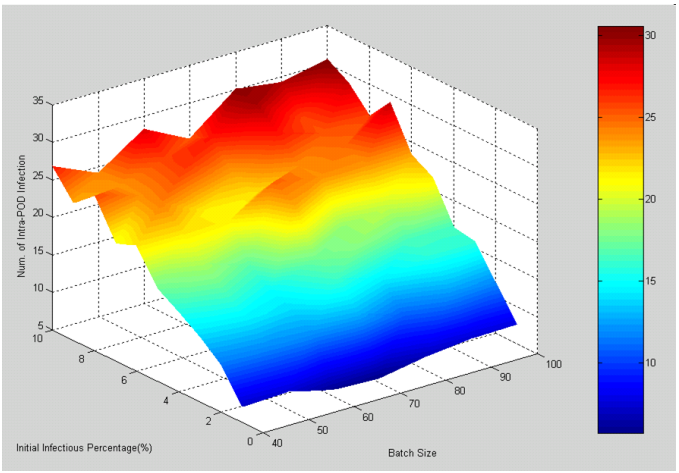


Figure 39 Number of intra-POD infections under different batch size and initial infectious percentage

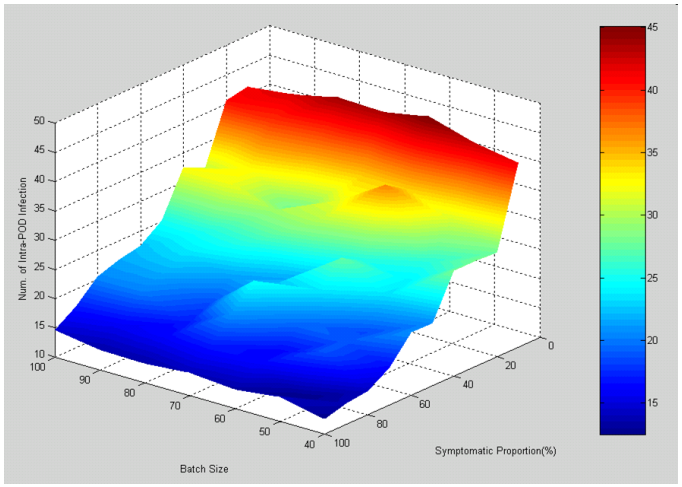


Figure 40 Number of intra-POD infections under different batch size and symptomatic proportion

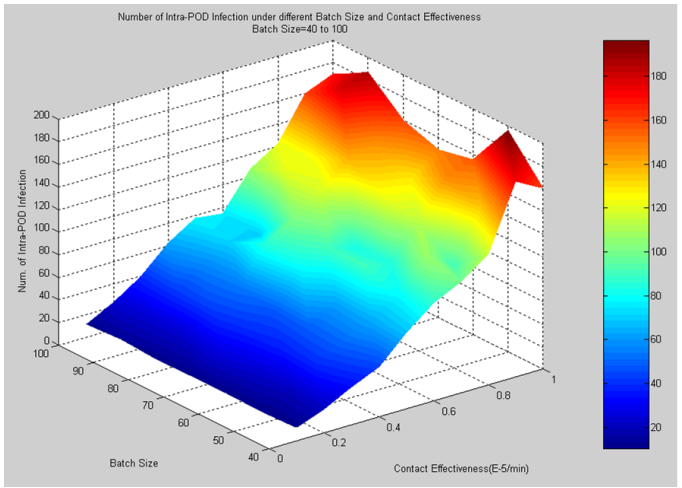


Figure 41 Number of intra-POD infections under different batch size and contact coefficient

3.3.5.4.4 Summary

Sensitivity analysis reveals that inclusion of a triage station, even when high accuracy cannot be guaranteed, is an effective and important approach for containing the intra-POD infection. The effect is more significant when disease outbreak is large-scale, disease symptomatic proportion is high, and the contact coefficient is high. In addition,

decentralization dispensing has also proven to be a better strategy than centralization dispensing. As indicated in [64], given various infectivity parameters and triage performance, there is an optimal throughput for operating PODs.

While it may seem counterintuitive, the sensitivity analysis studies show that increasing batch size within a threshold range does not affect intra-POD infections. In some scenarios the number of intra-POD infections is insensitive to batch size. This finding offers confidence in epidemiological safety regarding use of large batch sizes within POD operations, when the capacity allows. This offers improvement in operations as it lowers the client to staff ratio.

Our study confirms the importance of contact based disease propagation analysis. By incorporating contact based infection into both the ODE model and the CDC-RealOpt-POD simulation environment, we showed that the seamless integration of optimization, simulation, and disease propagation functions within CDC-RealOpt-POD offers higher applicability and better estimates in analyzing disease spread. The tradeoffs lie in the more rapid solution strategies (< 60 seconds) for the ODE system versus and the flexibility in stochastic service distribution input and slight increase in the CPU running time (within 3 minutes). Disease propagation module is embedded in the simulation and hence is run 10 times as well. Empowered by the fast computational time in CDC-RealOpt-POD, POD designers, managers, and epidemiologists can respond real-time to the changing conditions during the dispensing processes.

CHAPTER 4

INTEGRATION OF FLOW AND LAYOUT DESIGN

4.1 Receipt, Stage and Storage of Strategic National Stockpile

4.1.1 Background and Motivation

The public health emergencies can be either man-made, such as bioterrorist attacks, or natural disaster, such as pandemic outbreaks. During the outbreak of such emergency events, it is important to have clear guidelines for the receiving, staging, and storing of the distribution of critical supplies such as antibiotics, vaccines, medical supplies, and other items that can be used in the public health emergencies. All these items that are prepared for such purpose are procured, stored and managed by the Division of Strategic National Stockpile (SNS) at CDC. In particular, the SNS consists of two primary components: (a) the 12-hour Push Package, and (b) the Managed Inventory (MI). When there is a need for requesting the SNS, the State and the District must submit the request, then the corresponding transportation will be arranged to deliver the right type of SNS to Receiving, Staging and Storage (RSS) sites as soon as possible.

In the RSS sites, the stockpiles arrive in the form of aircraft shipping container. Each container is given a number and is color-coded for a specific type of medical countermeasure. All containers have same dimension for the base, but the height is different between a large container and a small container. The shapes of the containers are designed such that they utilize the space within an aircraft in the best way possible. All containers are wheeled for easy transportation.

When a push package is delivered to the RSS site (usually a warehouse), the containers are unloaded and positioned to designated areas according to its number.

These shipping containers are arranged in the RSS site as a matrix form. . The RSS site workers will pick required items from these containers based on picklists. Picklist is the list on which the demand of items for a Regional Distribution Node (RDN) is itemized. After picking, a simple sorting and packaging operation is conducted. The sorted packages are then loaded to trucks and delivered. .

In the RDN site, the delivered packages from RSS site will further be arranged, picked, sorted and packaged, and be delivered to Point-of-Dispensing (POD) where the large-scale mass dispensing is on-going for the general population. Figure 42 illustrates the process of delivering medical countermeasures to general population.

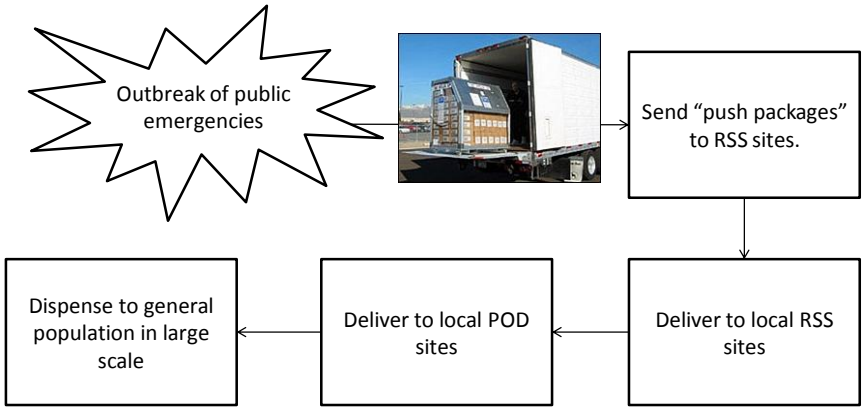


Figure 42 Distribution of strategic national stockpiles to general population

The operations in RSS site from stockpiles to delivering rearranged packages to RDN requires both the design and planning regarding to process flow design and layout design. For public health professionals, it is necessary to have a platform on which both functionalities are addressed and integrated seamlessly. This not only streamlines the planning process that involves authorities and professionals from different departments, it also facilitates the operations in the execution stage when modifications will have to be made on-the-fly to address the uncertainties from real situation. In this chapter, we propose a framework that integrates the process flow design and layout design.

As a functional extension of the original CDC-RealOpt-POD [60][61], we will discuss the implementation of RealOpt-RSS. RealOpt-RSS employees the CDC-RealOpt-POD modules, and include extra modules that integrate process flow and layout together. The system was employed in analyzing the design of guest flow and operations logistics for the Dolphin Tales in Georgia Aquarium.

4.1.2 RealOpt-RSS and Layout Design

Primary issue in the layout design is to have this function easy to use so that public health users can have a tool to quickly sketch the exterior shape and interior layout of the RSS site. Figure 43 shows how the layout can be represented in the system.

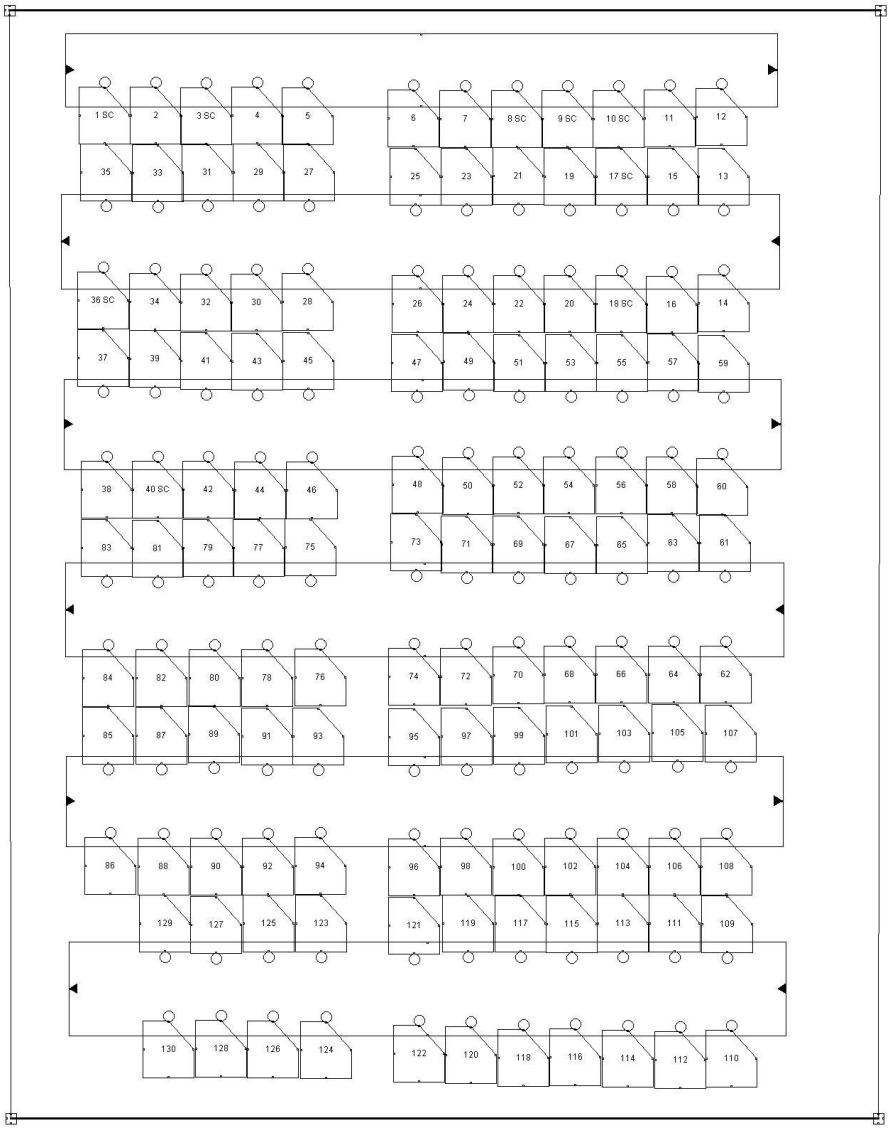
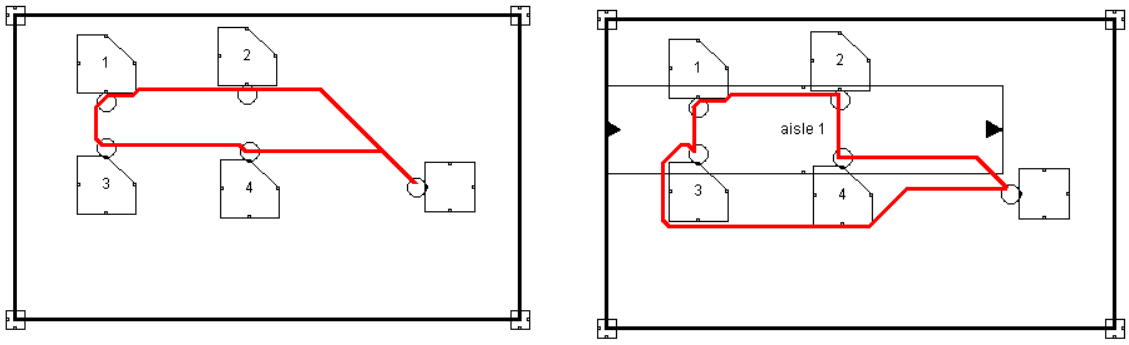


Figure 43 User-drawn RSS site layout in RealOpt-RSS

Users can put together such a layout easily. We provide object grouping and duplication function to facilitate the process of generating multiple almost identical objects. The pick-side of each object, in this example we used aircraft shipping container, can be customized as well. In addition, by observing the real RSS Exercise Event in 2010 we noticed that, similar to regular warehouse operations, in RSS site it is also necessary

to walking aisles as single-directional. In the system implementation we fully covered such requirement, as shown in Figure 44, to address the operations practically.



(a) without Aisle Direction Constraint (b) with Aisle Direction Constraint

Figure 44 Find picking path in user-drawn layout

4.1.3 RealOpt-RSS and Order Picking

The design of layout provides a platform for all the items in the stockpile to be located. Based on those locations we can incorporate picklists into the system and map them onto the user-drawn layout for visualization and potential process improvement. In this system, after defining a layout either from system-defined library or from users' drawing, users can import list of items—which usually summarized the name of items, supply quantity of items, and location of items—into the system, system will then automatically arrange these items onto the drawn layout according to provided information. A picklist can be imported, and system will search for the optimal picking route for this picklist. Figure 45 shows the integration of these information and functions in this system.

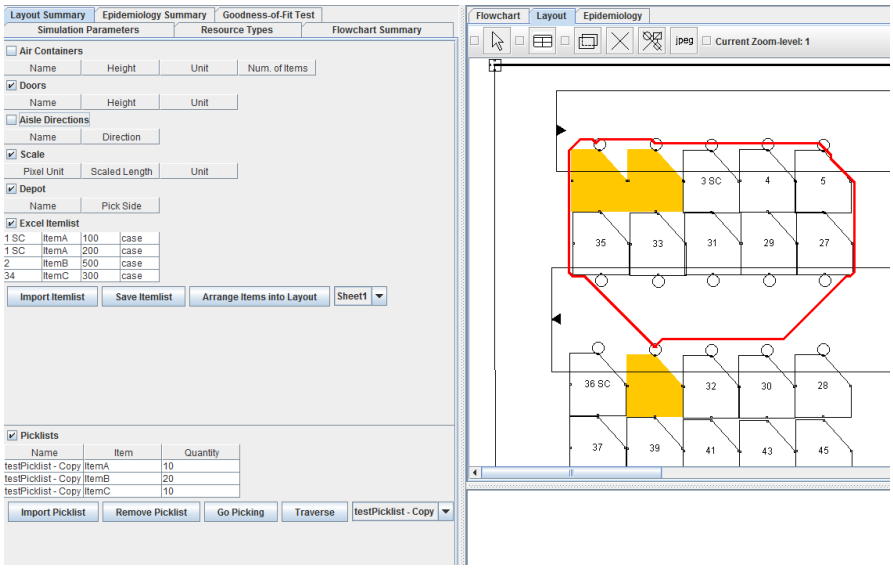


Figure 45 Integration of layout design, summarized item information, picklist, and picking route

4.1.4 RealOpt-RSS and Process Time

As mentioned in earlier section, the operations in RSS site primarily include unloading, staging, picking, sorting, packaging and loading. While the process time of some of these operations are irrelevant to the layout design, such as sorting and packaging, and some are minorly influenced by layout design, such as unloading and staging, there are operations, in particularly picking, whose process times are fully determined by the layout design and picking route as well. Based on the layout design and picking route mentioned in earlier section, it is possible to, while determining the process time for picking operation in process flow optimization, utilize the practical layout information for process flow optimization, and thus make the optimization results more realistic.

4.2 Designing Guest Flow and Operations Logistics for the Dolphin Tales

The content of this section consists of excerpts from a published paper [54].

4.2.1 Background and Motivation

The Georgia Aquarium is the world's largest aquarium with over 8.5 million gallons of water and 120,000 animals. On April 2nd 2011 the Georgia Aquarium opened a new exhibit, Dolphin Tales, dedicated to creating memorable connections between humans and animals and helping guest better understand one of mankind's favorite aquatic animals. To prepare for the new exhibit, it is necessary to design the exhibit's guest movement with the goal of enhancing the guest experience while limiting congestion.

4.2.2 Point of Interest

Data was collected primarily during a two-month period from August to October 2010. Both weekdays and weekend were included. Furthermore, our team tried to collect data from each of Aquarium's opening hours, as we observed that both types of data are time-dependent. For example, the inter-arrival time during 11:00 and 13:00 is usually higher, and during 12:00 to 13:00 the percentage of guests going to cafeteria is higher. The heterogeneity of inter-arrival times and transition probabilities is considered in our simulation model to capture the real situation.

In our simulation model, the aquarium is composed of 14 points-of-interest (POIs). These POIs include real exhibits, and other POIs that have to be included for completeness. For example, the Restroom in the main lobby is included because it is one of the primary destinations when customers leave exhibit.

Movements between different POIs are described by a transition probabilities table. This table quantifies the distribution of destinations after a customer leaves a POI as

origin. Under this circumstance, while it is common to use Decision block to connect different POIs and model the transition probabilities in ARENA, the disadvantage is that this model will be literally unreadable. Moreover, since the transition probabilities are time-dependent, (for example, the percentage of people going to Cafeteria from River Scout is a lot higher during 12:00 to 13:00 compared to 14:00 to 15:00) the heterogeneity in transition probabilities makes the explicit modeling of transition probabilities infeasible.

The data collected can be primarily classified into two types: (1) data regarding one POI, and (2) data regarding two POIs.

(1) This type of data includes customer arrival time at POI, customer departure time from POI, and visiting time within POI. Our team developed an Excel-based computer program to collect arrival and departure of POI, and from which inter-arrival times and inter-departure times can be derived. Our team designed a timestamp card to collect the visiting time within each POI.

Customers received a time-stamp card when they entered a POI (exhibit in particular). They returned the card to our team when they departed. On average, less than half of the cards dispensed were returned, since many customers simply forgot to return this card, or they preferred to keep it as a souvenir.

(2) This type of data includes the transition probability between POIs. We randomly selected customers and recorded their origin and destination, for example, from River Scout to Cafeteria. After aggregating these data by origin, we can get the distribution of destinations from that origin. In addition, the transition times between POIs were collected too since a timestamp was automatically collected along with the origin/destination.

4.2.3 Agent-based Simulation

Considering the system characteristics in Georgia Aquarium simulation, in this study we aim to develop an agent-based simulation system that assists in real-time process planning and layout design, and system optimization such as minimizing resource requirements, maximizing throughput, and manual resource reallocation.

When it comes to modeling individual behavior in a system, the discrete event simulation (DES) [48] is in most cases used in systems that are better described in a "top-down" manner. More specifically, when individuals are required to go through a series of predefined processes (these can be services, works, or anything that can trigger status change on individuals for a period of time) that describes the primary system characteristics, it is more straightforward to model this system from the "process" point of view.

However, in this Georgia Aquarium project which shares similar properties from other Theme Park crowd control researches [18][39], the individuals are necessary to be modeled by "autonomous entities" that can actively make decision, instead of being passively guided by the system-defining process flow. The Agent-based simulation (ABS) [30] offers a "bottom-up" modeling philosophy that suites the autonomy of individuals better.

In this study we develop a system, called RealOpt-ABM, which can in fact tackle a more generalized problem structure. While it is capable of modeling a pure flow-based system such as a point-of-dispensing (POD) for dispensing medical countermeasures [56][60][61][62], and modeling a theme park where individuals are fully autonomous in determining where to go, our tool is capable of modeling a system in which individual behaviors are *hybrid*—part of their behaviors are externally described by a series of

predefined processes, and part are internally determined by individuals themselves. The switching between two parts of individual behaviors can be either time-dependent or location-dependent.

4.2.4 Integration of Process Flow and Layout Design

Compared to other multi-agent simulation libraries, such MASON [69], NetLogo [88], Swarm [75], and Repast [13], as a simulation system, in RealOpt-ABM we emphasize the ability to facilitate and streamline the model building process, and help users that are not familiar with mathematical programming and computer simulation to build a complicated model. We enable users to describe a model by graphically defining process flow and physical layout. Other underlying mechanisms in agent-based simulation, such simulated environment/map construction, path planning, collision detection and avoidance, and agent management, are encapsulated and handled inside the system. This allows users to focus on exerting their domain knowledge in building a model, instead of working on building a computer program. This shortens the cycle time in model analysis, and reduces the possible workload in both model verification and validation as well.

In the system front-end, RealOpt-ABM provides Flowchart drawing panel for defining process flow, and Layout drawing panel for designing physical layout.

4.2.4.1 Flowchart Panel

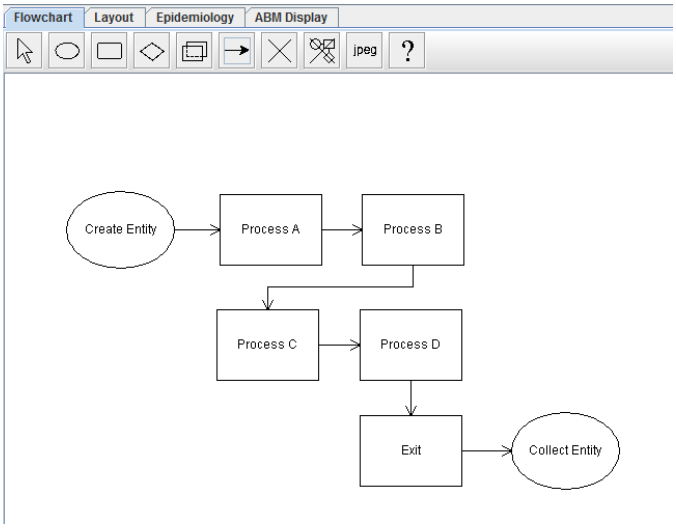
In the Flowchart drawing panel, user defines processes in a system, and precedence relations between those processes. In RealOpt-ABM we allow users to model the precedence constraints at three different levels:

- on all the processes. For each process, its successive process is predefined. The dispensing of medical countermeasures in points-of-dispensing (PODs) falls into this category.

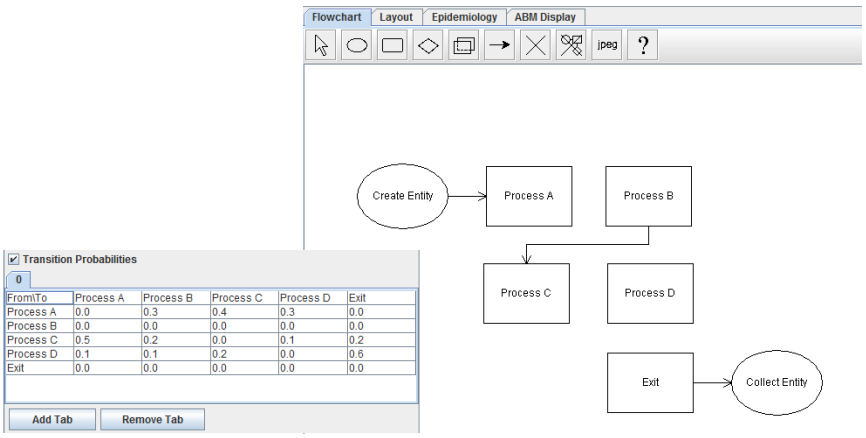
- on part of the processes. Part of the processes has predefined successive processes, and others are assigned with successive processes dynamically during the progress of simulation by either probabilistic distribution, or by agent's discretion. Medical examination in hospital falls into this category.

- no precedence constraints. All the processes are assigned with successive processes dynamically during the progress of simulation by either probabilistic distribution or by agent's discretion. Traversing between exhibits in a theme park or Aquarium falls into this category.

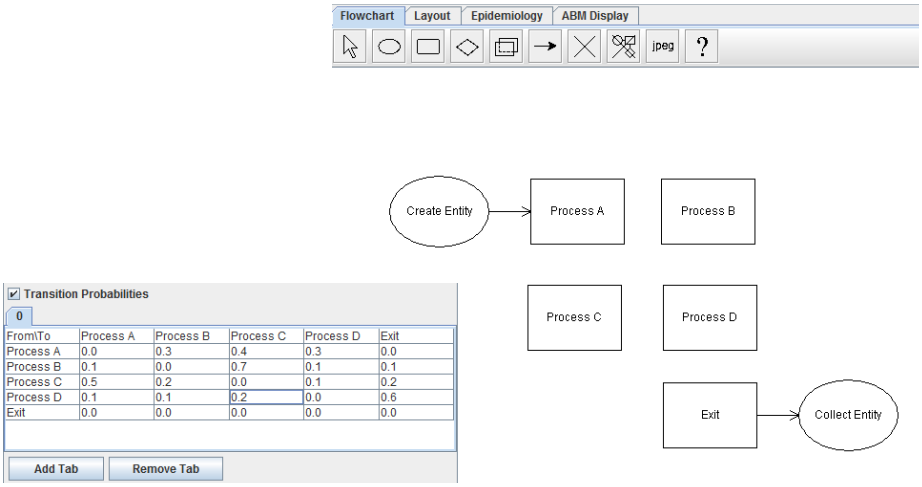
Figure 46 illustrates building process flow under different precedence constraints using RealOpt-ABM:



(a) Precedence Constraints on All Processes



(b) Precedence Constraints on Part of Processes

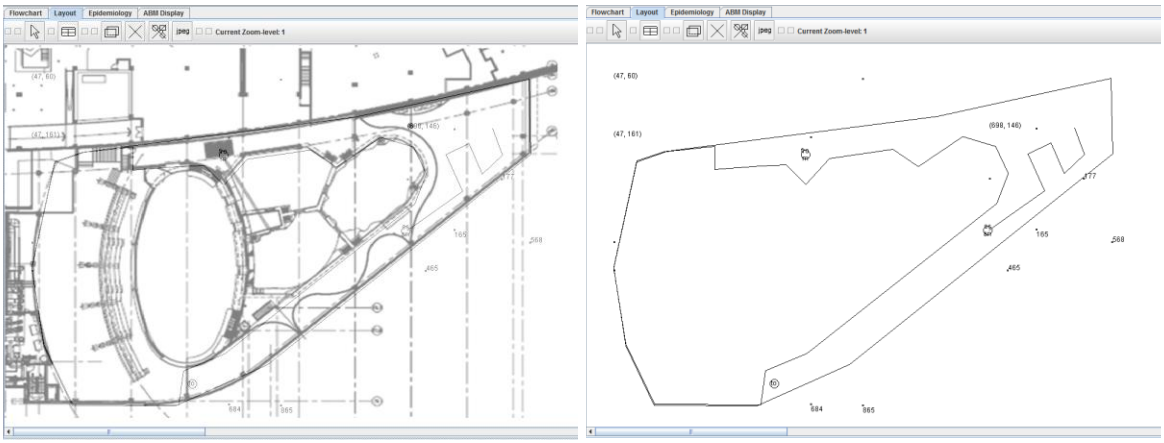


(c) No Precedence Constraints

Figure 46 Building process flow under different precedence constraints

4.2.4.2 Layout Panel

The Layout drawing panel allows users to combine a set of simple polygons, convex or concave, to represent the physical layout of a simulated facility. Users can choose to overlay a picture as the real layout on the panel to facilitate the drawing process, or start designing a layout from scratch. Figure 47 shows the drawn layout of the Dolphin Lobby in Aquarium with and without overlaid picture.



(a) with Overlaid Blueprint (b) without Overlaid Blueprint

Figure 47 The drawn layout of Georgia Aquarium Dolphin Lobby in RealOpt-ABM [54]

4.2.4.3 Integration of Process Flow and Physical Layout

Process flow and physical layout are integrated by simply assigning each process with a location/POI, on the physical layout. Figure 48 illustrates two models with identical process flow and physical layout, but different spatial flow.

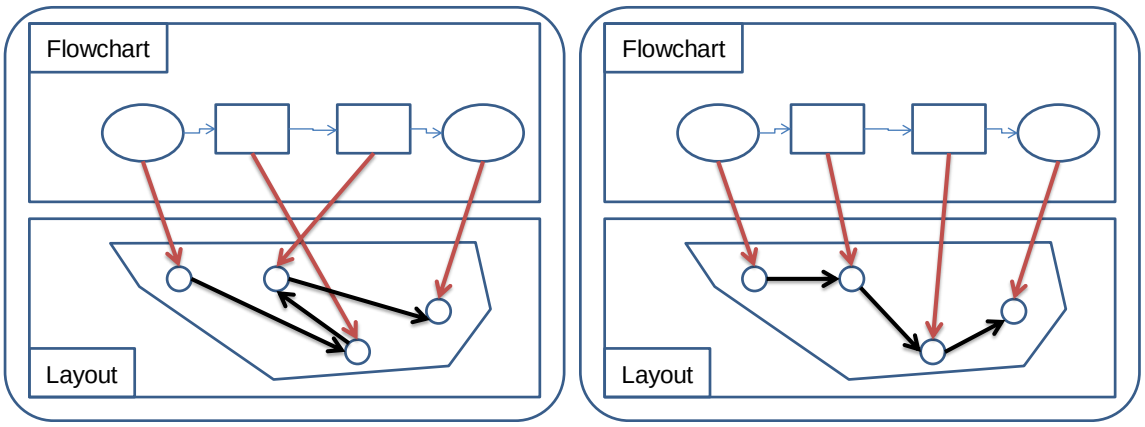


Figure 48 Illustrative integration of process flow and physical layout [54]

While it is an option to allow users define process flow directly on the physical layout, we choose to design the system in such an indirect way that process flow and physical layout are designed separately for several reasons:

- This provides the flexibility that process flow can be modeled individually without having physical layout, since the spatial properties are not always required for process optimization/simulation.
- This improves the readability of both Flowchart panel and Layout panel.
- Most importantly, this design parameterizes the spatial flow of agents in simulation and thus facilitates the sensitivity analysis in investigating space usage and crowd control.

As for system implementation, an object Flowchart is declared for the manipulation of process flow on Flowchart panel, and another object Layout is declared for the manipulation of physical layout on the Layout panel. Object Flowchart contains sub-objects such as Process to represent each individual process. And object Layout contains sub-objects such as FlowNode to represent potential location for processes. Users can assign FlowNode to corresponding Process to map process flow onto spatial flow.

In fact, there are four types of sub-objects that are capable of being assigned with a location on the physical layout. They are:

- Create block: This object creates entities/agents that enter the simulation, and is usually used for modeling entrance of a simulated facility. Entities created can be households or individuals that are streaming into the simulated facility.
- Process block: This object represents activity in a simulation that can trigger status change on individuals for a period of time by consuming resource or resources. Typically it is used for modeling services in a simulated facility.

For example, service can be triage in a POD, blood test in a hospital, or ticket scanning in Aquarium.

- Delay block: This object is basically identical to Process block, except for that Delay block does not consume resources and therefore there will be no waiting line formed in a Delay block.

- Dispose block: The object collects and removes entities/agents from the simulation, and is usually used for modeling exit of a simulated facility.

4.2.5 Local Path Finding

In local path planning, the drawing of each region on Layout panel is discretized into cells, and these cells are used as nodes to construct a graph. A pair of nodes is connected by an undirected edge if corresponding cells are mutually accessible. Inaccessibility occurs, for example, when one cell is located inside an internal obstacle in this region, or outside the external boundary of this region. By applying shortest path algorithm [1], a path can be obtained between any feasible origin-destination pair. Notice that the origin-destination pair in starting region is global origin-connector, in intermediate region it is connector-connector, and in ending region it is connector-global destination.

Considering the problem size in solving the local path planning problem, the local path planning is pre-solved before launching simulation, and all the feasible paths are available for quick reference during the progress of simulation.

4.2.6 Global Path Finding

4.2.6.1 Global Path Finding with Full Information

In Georgia Aquarium, while most of the exhibits are located on the 1st floor, two paid exhibits (Deepo 4D Show and Planet Shark) are set up on the 2nd floor. Furthermore, the Dolphin Lobby is built outside the main body of the Aquarium and is connected to the main lobby on 1st floor by escalator and elevator. This shows the situation where multiple regions are incorporated in one simulated environment.

Although easy to trace in the Aquarium example, the connectivity can be complicated in general case. Figure 49 illustrates a scenario where leaving a building from 4th floor requires sequentially walking through 2nd floor and 3rd floor, and walking to 2nd floor from 3rd floor takes 1st floor as intermediate region.

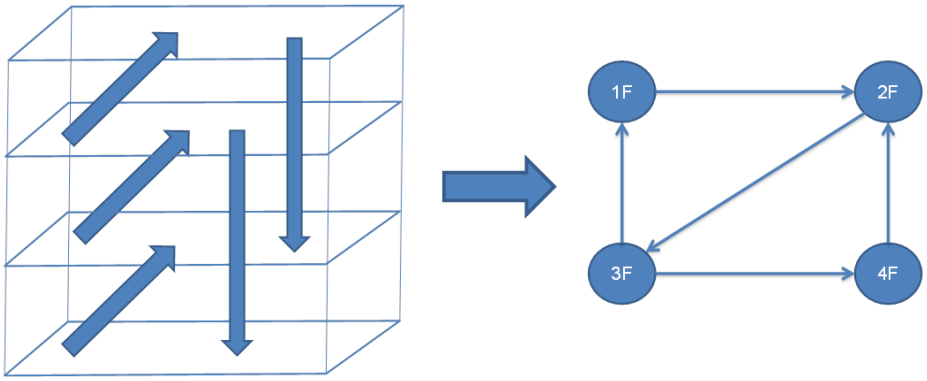
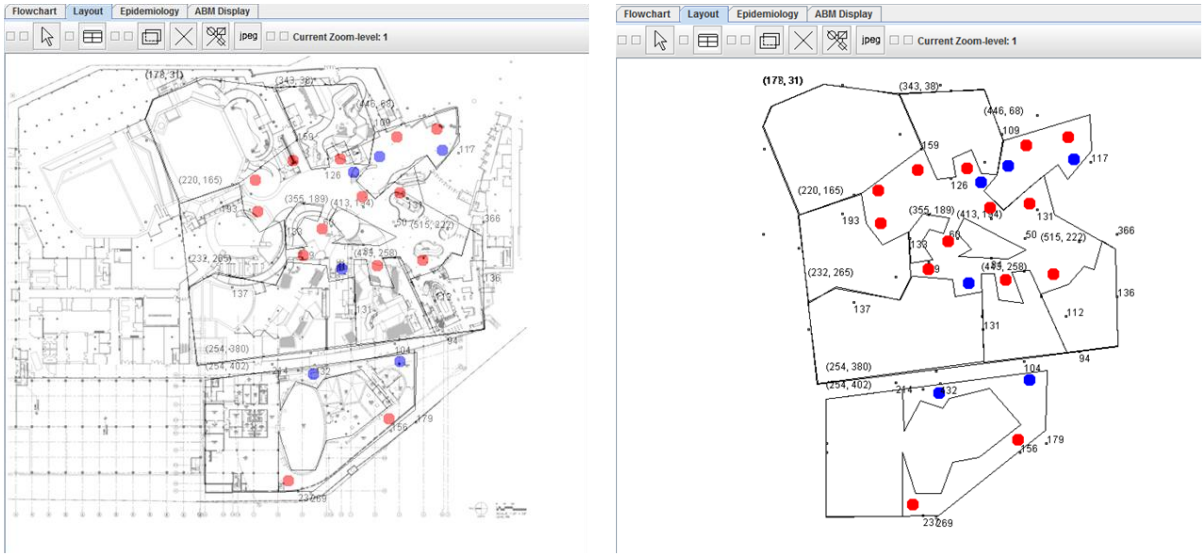


Figure 49 Illustrative multi-floor building and corresponding connectivity graph

Considering the concept of topological map which represents topological relations between different parts of the virtual world in Shao [84], and spatial subdivision in Lamarche [51], RealOpt-ABM implements the idea by providing the sub-object Connector on Layout panel to define the connectivity between disjoint regions for modeling a system with multiple regions, and automatically generates the connectivity graph from user's drawing.

During path finding, the physical layout, with Connectors, are transformed into a connectivity graph, in which nodes in the connectivity graph represent "regions" in the original layout, and pair of nodes is connected by a directed arc if corresponding regions in original layout are connected by Connectors.

Figure 50 shows the actual Georgia Aquarium model in RealOpt-ABM as physical layout. For attractions, only the actions inside dolphin area are modeled in detail. For other attractions, such as River Scout or Coldwater Quest, it is simplified to a Delay. This model includes three different regions, the 1st floor of aquarium, the 2nd floor of aquarium, and dolphin area, and they are connected by Connectors. Blue dots represent Connectors, and red dots represent FlowNodes. FlowNodes are where Process Block or Delay Block in the Flowchart are assigned to. That means, customers are "processed" (in this case they are viewing the attractions) in those locations. Figure 51 shows the transformed connectivity graph of the Georgia Aquarium model.



(a) With background blueprint (b) Without background blueprint

Figure 50 Georgia Aquarium model as physical layout with Connector and FlowNode [54]

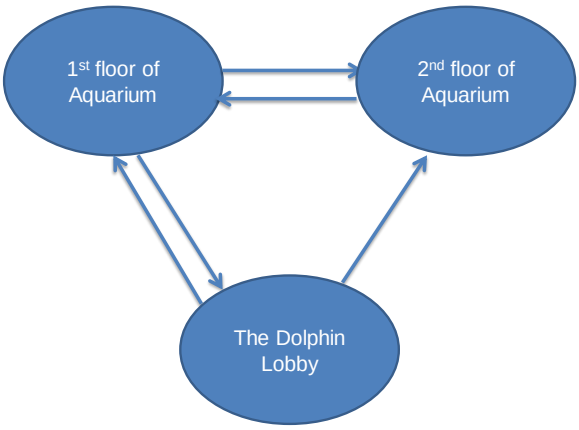


Figure 51 Connectivity graph of Georgia Aquarium model

The path planning in such environment requires both global path planning and local path planning. In global path planning, shortest path algorithm [1] is applied on the connectivity graph to identify the sequence of intermediate regions that an agent needs to walk through to reach its destination.

After identifying the sequence of regions to walk through, the path planning is complete by sequentially conducting local path planning within each region.

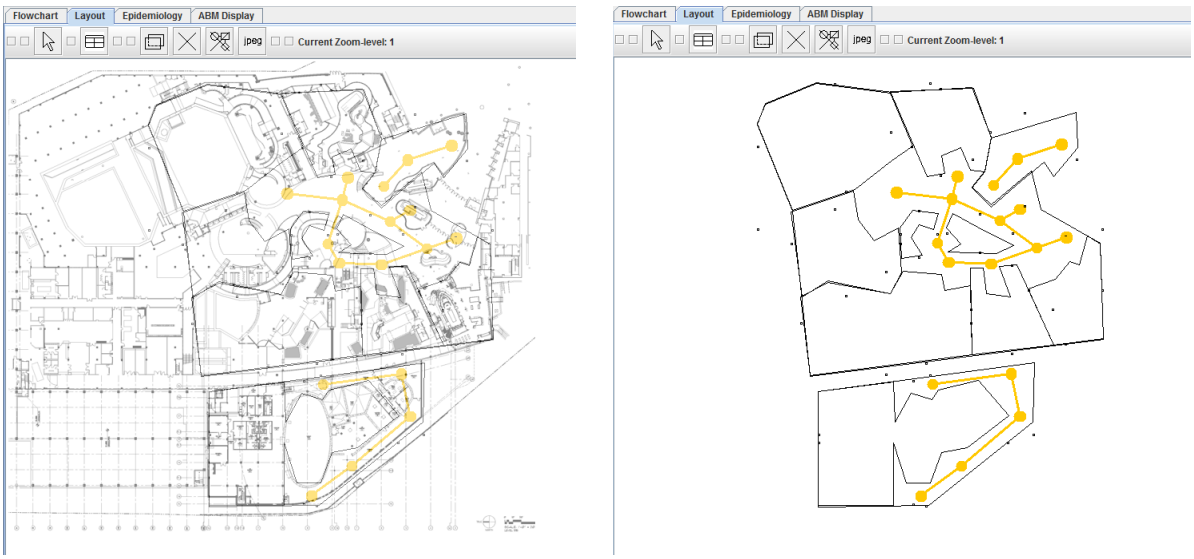
4.2.6.2 Global Path Finding with Partial Information

With full knowledge of the layout in a simulated environment, the path finding is implemented in two layers: global path planning and local path planning. Based on the characteristics of transformed graph (for example, directed or undirected graph), different shortest path algorithms, such as Dijkstra algorithm or A* (A-star) algorithm, can be employed for different purposes.

However, when the full knowledge of the layout in simulated environment is not available, a roadmap [6][53][79] is necessary for identifying the critical points for path planning. Notice that when the simulated environment has multiple regions, each of those regions requires its own roadmap.

Considering the situation where no information about the simulated environment is available to agent, agent will have to conduct depth-first-search (DFS) on those critical points that are identified from the roadmap. The traversing path will form a tree. When partial information about the simulated environment is available, agent can selectively remove part of the nodes and edges connected to these nodes from the roadmap. Removal of nodes and edges implies agent's ability in recognizing unnecessary traversing regarding to the current destination.

RealOpt-ABM allows users to manually define roadmap for simulated environment by intuitively dropping draggable nodes and drawing edges on the layout, or modify a roadmap generated based on generalized Voronoi diagrams [49]. Figure 52 illustrates possible roadmap of Georgia Aquarium. When traversing, users can define agents' traversing node based on roadmap, or based on critical points only. Connectivity of roadmap will be automatically checked if necessary. Option is also provided to continuously detect critical points with better potential in reaching destination. More specifically, during backtracking, agent can skip backtracking to the most recent node and proceed to other detected unexplored node. Agents can appear more normal in their movements, however the running will be significantly slower due to the additional detecting.



(a) With background blueprint (b) Without background blueprint

Figure 52 Georgia Aquarium model as physical layout with roadmap

4.2.7 Agent-based Simulation

Considering the system characteristics in Georgia Aquarium simulation, in this study we aim to develop an agent-based simulation system that assists in real-time process planning and layout design, and system optimization such as minimizing resource requirements, maximizing throughput, and manual resource reallocation.

When it comes to modeling individual behavior in a system, the discrete event simulation (DES) [48] is in most cases used in systems that are better described in a "top-down" manner. More specifically, when individuals are required to go through a series of predefined processes (these can be services, works, or anything that can trigger status change on individuals for a period of time) that describes the primary system characteristics, it is more straightforward to model this system from the "process" point of view.

However, in this Georgia Aquarium project which shares similar properties from other Theme Park crowd control researches [18][39], the individuals are necessary to be modeled by "autonomous entities" that can actively make decision, instead of being passively guided by the system-defining process flow. The Agent-based simulation (ABS) [30] offers a "bottom-up" modeling philosophy which suites the autonomy of individuals better.

In this study we develop a system, called RealOpt-ABM, which can in fact tackle a more generalized problem structure. While it is capable of modeling a pure flow-based system such as a point-of-dispensing (POD) for dispensing medical countermeasures [56][60][61][62], and modeling a theme park where individuals are fully autonomous in determining where to go, our tool is capable of modeling a system in which individual behaviors are hybrid—part of their behaviors are externally described by a series of predefined processes, and part are internally determined by individuals themselves. The switching between two parts of individual behaviors can be either time-dependent or location-dependent.

While individual movement can be either pre-defined (by following path) and/or autonomous (by self-discretion), the move cannot violate the constraints imposed by physical layout.

The physical layout is basically the drawing drawn by users where individuals are allowed to move. Conceptually, the constraints are imposed as the following. First of all, the drawing is discretized into cells, and these cells are used as nodes in constructing a graph G. The edges of this graph are then added between corresponding pairs of nodes only when nodes in a pair are accessible to each other. By applying Dijkstra's Shortest Path Algorithm [1] to G, we can get a path, feasible under internal obstacles and external boundary, with any origin-destination.

When individual detours from pre-determined path and moves autonomously, limitations from internal obstacles and external boundary are still applied and feasibility is checked at each iteration.

In addition to the hybrid individual behaviors that RealOpt-ABM is capable of modeling, other primary system advantages are as the following:

- Graphical user interface: In RealOpt-ABM users are able to model a complicated system by simple drawing. Several drawing panels are provided to define important system characteristics such as process flow and physical layout. After integrating inputs as graphical form between these panels, the process flow can be simulated by taking spatial properties into consideration.

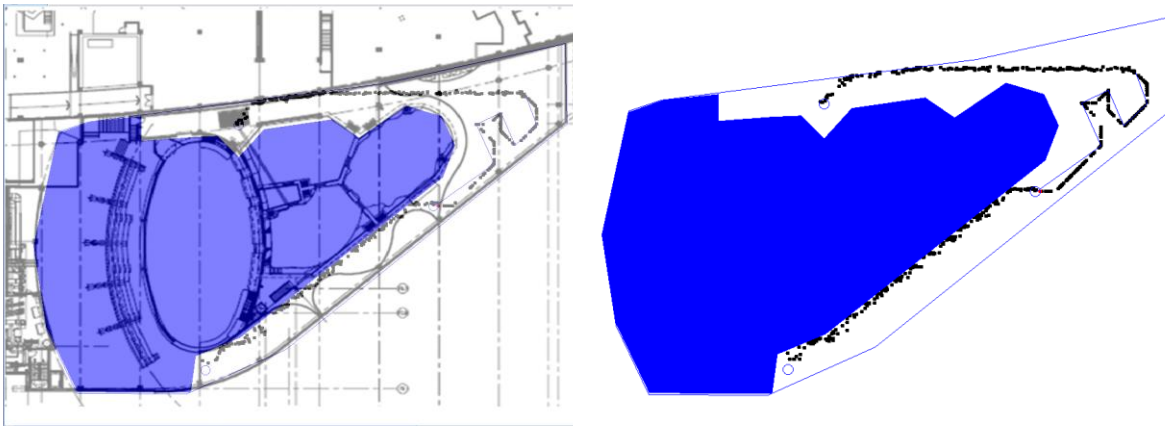
- Computational efficiency: the computational efficiency of RealOpt-ABM is contributed from two parts:

    - In optimization: By inheriting the computation capability from CDC-RealOpt-POD, a large-scale optimization and simulation decision support system [60][61], we are able to minimize the resource requirements, optimize resource allocation and maximize throughput allocation. This enables our system for strategic and operational planning as well as on-the-fly dynamic changes.

    - In agent-based simulation: We extended MASON [69], a multi-agent simulation library core in Java, as our agent-based simulation engine in RealOpt-ABM. While there are many open-sourced agent-based simulation libraries available, such as NetLogo [88], Repast [13], and Swarm [75], the choice of MASON primarily lies in its advantages in intensive computation and portability. As output, the modularity of

MASON animation also makes it better integrated with the input drawing modules from CDC-RealOpt-POD.

Figure 53 (a) and Figure 53 (b) show a snapshot of the loading process in the Dolphin area lobby. In Figure 53(a) the real layout of Dolphin area lobby is overlaid on the display panel for better understanding, while in Figure 53(b) only the drawn layout is shown for faster animation. Both figures capture the hybrid individual behaviors in the theater loading process. In particular, when individuals are moving along the predefined path, the randomness in human movement is also incorporated by allowing agent make autonomous (but limited) decision.



(a) With background blueprint (b) Without background blueprint

Figure 53 Layout of loading process in Dolphin Area under agent-based simulation [54]

In this example, customers are expected to proceed to the ticket scanning station after arriving the Dolphin area lobby. After ticket scanning, they are walking along the corridor and enter the Dolphin Theater. This process is defined as a flowchart in RealOpt-ABM, as shown in Figure 54. Parameters are input via the popup dialog after clicking on corresponding block in the flowchart.

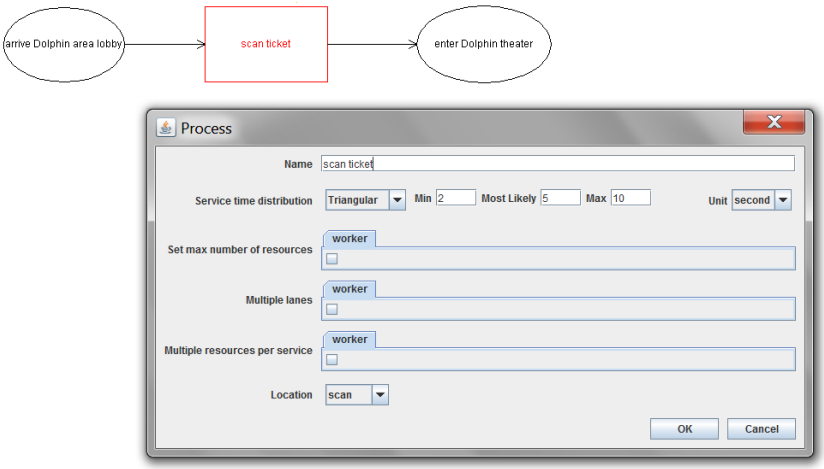
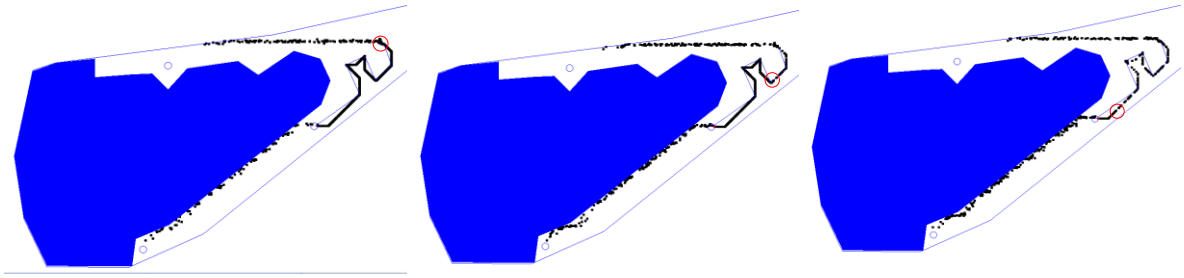


Figure 54 Process flow of loading process in Dolphin Area under agent-based simulation

In this study, while the number of scanners is required to be part of the input parameters in most simulation package, RealOpt-ABM provides the capability in optimizing (minimizing in this case) the number of scanners such that statistics such as waiting time and queue length are feasible. By comparing results under different assignment of scanners, In Figure 55(a), Figure 55(b), and Figure 55(c) it shows the queue length under different assignment of scanners, and this validates the importance of optimizing resource allocation for improving operations and enhancing customers experience as well.



(a) Number of Scanners = 1 (b) Number of Scanners = 2 (c) Number of Scanners = 3* (**optimal**)

Figure 55 Lengths of waiting line in Dolphin Lobby ticket scanning process

4.2.8 Integration of Process Flow, Physical Layout, Simulation, and Optimization

On the front-end, the process flow and physical layout are defined by user's drawing of flowchart and layout design as graphical input. After system running is launched, on the back-end, the drawings are interpreted and transformed into optimization model. Our optimization engine, which is designed primarily based on fluid model and minimum-cost flow network, solves the optimization model as minimizing resources allocation problem, or maximizing throughput problem.

The simulation model, after the optimization is run to completion, is constructed on the back-end by mapping the imported parameters from the optimization engine (such as number of workers assigned or achievable throughput) and transformed user's drawings. The simulation engine, which is extended from MASON [69], is subsequently launched based on the simulation model. More specifically, the physical layout is placed on the SparseGrid2D field, and process flow is mapped onto part of agent's behavior set.

Such system design maintains the flexibility of extending RealOpt-ABM from PC version to web-based version under client-server structure. While the optimization engine can always be kept on server to relieve users from intensive computation, the MASON-based simulation engine can be arranged either on the back-end for faster simulation speed, or on the front-end for visualization of simulation results.

CHAPTER 5

CONCLUSION AND FUTURE RESEARCH

This research focuses on mass dispensing in public health and emergency response. The research focuses on developing mathematical models, real-time algorithms, and computerized decision support systems that enable **(**1) systematic coordination to tackle multifaceted nature of mass dispensing, (2) fast disease propagation module to allow immediate mitigation response to on-site uncertainties, and (3) user-friendly platform to facilitate modeling-solution integration and cross-domain collaboration. The work translates operations research methodologies into practical decision support tools for public health emergency professionals.

The major mathematical contribution involves a novel 6-stage mathematical model that integrates dispensing operations within a disease propagation framework; and its generalization into a fundamental analytic framework. Our work is the first mathematical model developed that focuses on the interactions of dispensing operations and disease propagation inside the mass-dispensing clinic, and provide insight on designing the mitigation strategies. Computational and systems advances include the novel feasible genetic algorithm for difficult facility location problems; its integration into a real-time web-based decision support system; and the integration of process flow and layout design for receive, stage, and storage logistics operations.

Specifically, under the framework of modeling and optimizing the public health infrastructure for biological and pandemic emergency responses, the task first determines adequate number of point-of-dispensing sites (POD), by placing them strategically for best possible population coverage. Individual POD layout design and associated staffing are then optimized to maximize throughput and/or minimize resource requirement for an

input throughput. Large-scale dispensing creates a large influx of individuals to dispensing facilities, thus raising the risk of high degree of intra-facility infections. Our work characterizes the interaction between POD operations and disease propagation.

Fast genetic algorithm-based heuristics were developed for solving the integer-programming-based facility location instances. The approach has been applied to the metro-Atlanta area with a population of 5.2 million people over 11 districts. Among the 2,904 instances, the state-of-the-art specialized integer programming solver solved all except one instance to optimality within 300,000 CPU seconds and solved all except 5 to optimality within 40,000 CPU seconds. The fast heuristic algorithm returns good feasible solutions that are within 8 percent to optimality in 15 minutes. This algorithm was embedded within an interactive web-based decision support system, RealOpt-Regional. The system allows public health users to contour the region of interest and determine the network of PODs for their affected population. Along the fast optimization engine, the system features geographical, demographical, and spatial visualization that facilitate real-time usage. The client-server architecture facilities front-end user interactive design on Google Maps© while the facility location mathematical instances are generated and solved in the back-end server.

In the analysis of disease propagation and mitigation strategies, we first extended the 6-stage ordinary differential equation-based (ODE) compartmental model to accommodate POD operations. This allows us to characterize the intra-facility infections of highly contagious diseases during local outbreak when large dispensing is in process. The disease propagation module was then implemented into the CDC-RealOpt-POD discrete-event-simulation-optimization. CDC-RealOpt-POD is a widely used emergency response decision support system that includes simulation-optimization for determining optimal staffing and operations. We analyzed the interactions between POD operations and disease parameters and identified effective mitigation strategies. The disease

propagation module allows us to analyze the efficient frontier between operational efficiencies and intra-POD infections. Emergency response POD planners and epidemiologists can collaborate under the familiar CDC-RealOpt-POD environment, e.g., to design the most efficient plan by designing and analyzing both POD operations and disease compartmental model in a unified platform. Corresponding problem instances are formed automatically by combining and transforming graphical inputs from users.

To facilitate the operations of receiving, staging and storage (RSS) of medical countermeasures, we expanded the CDC-RealOpt-POD layout design functions by integrating it with the process flow. The resulting RSS system allows modeling of both system processes along with spatial constraints for optimal operations and process design. In addition, agent-based simulation was incorporated inside where integrated process flow and layout design allow analysis of crowd movement and congestion. We developed the hybrid agent behavior where individual agents make decision through system-defined process flow and autonomous discretion. The system was applied successfully to determine guest movement strategies for the new Georgia Aquarium Dolphin Tales exhibit. The goal was to enhance guest experience while mitigating overall congestion.

On possible future research direction lies in methodological and algorithmic advances for incorporating the mathematical formulation developed in disease propagation into a large-scale simulation-optimization framework for further analysis of the efficient frontier between operational efficiencies and intra-POD infections. Based on the current research, this area remains challenging and important. There remains many modeling and computational opportunities. Tackling disease propagation is extremely computational expensive via simulation, and it is numerically unstable through numerical approximation. More research has to be performed investigate theoretical and computational advances of approximation algorithms. Furthermore, the compartmental model itself can be extended to incorporate additional characteristics such as spatial

proximity. This will fit into the individual-based epidemiology investigation. User-drawn layout design can too be integrated into disease propagation analysis. In addition, a better local search framework with different neighborhood definition and different warm-start strategy can be proposed for the local search-based optimization for improving solution quality and convergence.

Another possible direction can be focused on developing methodologies and computerized systems for real-time event awareness. The integrated architecture of RealOpt-Regional and CDC-RealOpt-POD facilitate efficient planning and execution of large-scale dispensing. However, it remains a great challenge to detect the outbreak as early as possible. Trending a potential disease outbreak is an important problem and has become a popular area of research by social and computer scientists. More research can be performed in advancing methodologies and decision support systems for real-time event awareness. Social networking media such as Twitter, Facebook, and Google+ provide rich data sources for event awareness. Google has employed the massive online data to identify flu trends across the nation. Other online sources such as newspapers and news channels may be incorporated as well. The challenge here involves large-scale heterogeneous data sources. Extracting useful information with data heterogeneity poses another challenge. Topic modeling has been employed effectively in large-scale data mining and in medical domain but will offer limited usage when dealing with data streams such as Tweets. Advances in machine learning and natural language processing are expected. Web-based systems remain essential as they allow real-time usage and user interaction.

REFERENCES

[1] Ahuja RK, Magnanti TL, Orlin JB (1993), Network Flows: Theory, Algorithms, and Applications, Prentice Hall.

[2] Aickelin U (2002), An indirect genetic algorithm for set covering problems. Journal of the Operational Research Society, 53(10), pp 1118-1126.

[3] Alibek K, Handelman S (2000), Biohazard: The Chilling True Story of the Largest Covert Biological Weapons Program in the World—Told from Inside by the Man Who Ran It. Random House. New York.

[4] Anderson RM, May RM, Anderson B (1992), Infectious diseases of humans: dynamics and control. Oxford University Press.

[5] Bellamy RJ, Freedman AR (2001), Bioterrorism, QJM: An International Journal of Medicine, 94(4), pp 227-234.

[6] Boor V, Overmars MH, van der Stappen AF (1999), The Gaussian sampling strategy for probabilistic roadmap planners, Proceedings of the 1999 IEEE International Conference on Robotics and Automation, 2, pp 1018-1023.

[7] Bravata DM, McDonald KM, Smith WM, Rydzak C, Szeto H, Buckeridge DL, Haberland C, Owens DK (2004), Systematic review: Surveillance systems for early detection of bioterrorism-related diseases, Annals of Internal Medicine, 140(11), pp 910-922.

[8] Byrka J (2007), An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. Proceedings of the 10th International Workshop on Approximation and the 11st International Workshop on Randomization, and Combinatorial Optimization. Algorithms and Techniques, pp 29-43.

[9] Centers for Disease Control and Prevention (CDC) (2003), Maxi-Vac Version 1.0, Software available at http://www.bt.cdc.gov/agent/smallpox/vaccination/maxi-vac/.

[10] Charikar M, Guha S (1999), Improved combinatorial algorithms for facility location and k-median problems. Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science, pp 378–388.

[11] Chudak FA, Shmoys D (2003), Improved approximation algorithms for the uncapacitated facility location problem. SIAM J. Comput, 33(1), pp 1–25.

[12] Clizbe JA (2004), Challenges in managing volunteers during bioterrorism response, Biosecurity and Bioterrorism: Biodefense Strategy, Practice, and Science, 2(4), pp 294-300.

[13] Collier N (2001), Repast: An agent based modeling toolkit for Java, http://repast.sourceforge.net.

[14] Das TK, Savachkin AA, Zhu Y (2008), A large-scale simulation model of pandemic influenza outbreaks for development of dynamic mitigation strategies, IIE Transactions, 40(9), pp 893-905.

[15] Dibble C, Wendel S, Carle K (2007), Simulating pandemic influenza risks of US cities, Proceedings of the 2007 Winter Simulation Conference, pp 1548-1550.

[16] Diekmann O, Heesterbeek JAP (2000), Mathematical epidemiology of infectious disease: model building, analysis and interpretation, John Wiley and Sons.

[17] Drenkard K, Rigotti G, Hanfling D, Fahlgren TL, LaFrancois G (2002), Healthcare system disaster preparedness, Part 1: Readiness planning, Journal of Nursing Administration, 32(9), pp 461-469.

[18] Dubiel B, Tsimhoni O (2005), Integrating agent-based modeling into a discrete event simulation, Proceedings of the 2005 Winter Simulation Conference, pp 1029-1037.

[19] Edmonds J, Karp RM (1972). Theoretical improvements in algorithmic efficiency for network flow problems, Journal of the ACM, 19(2), pp 248-264.

[20] Epstein J, Cummings DAT, Chakravarty S, Singa R, Burke DS, Cummings JD (2004), Toward a Containment Strategy for Smallpox Bioterror: An Individual-Based Computational Approach, The Brookings Institution Press, Washington, D.C.

[21] Eubank S (2002), Scalable, efficient epidemiological simulation, Proceedings of the 2002 ACM symposium on Applied computing, pp 139-145.

[22] Fauci AS (2002), Smallpox vaccination policy-The need for dialogue, The New England Journal of Medicine, 346(17), pp 1319-1320.

[23] Ferguson NM, Cummings DA, Cauchemez S, Fraser C, Riley S, Meeyai A, Iamsirthaworn S, Burke DS (2005), Strategies for containing an emerging influenza pandemic in Southeast Asia, Nature, 437, pp 209-214.

[24] Ferguson NM, Cummings DA, Fraser C, Cajka JC, Cooley PC, Burke DS (2006), Strategies for mitigating an influenza pandemic, Nature, 442, pp 448-452.

[25] Franz DR, Jahrling PB, Friedlander AM, McClain DJ, Hoover DL, Bryne WR, Pavlin JA, Christopher GW, Eitzen EM Jr (1997), Clinical recognition and management of patients exposed to biological warfare agents, The Journal of the American Medical Association, 278(5), pp 399-411.

[26] Fraser C, Riley S, Anderson RM, Ferguson NM (2004), Factors that make an infectious disease outbreak controllable, PNAS, 101(16), pp 6146-6151.

[27] Gani R, Leach S (2001), Transmission potential of smallpox in contemporary populations, Nature, 414(6865), pp 748-751.

[28] Gen M, Cheng R (1997), Genetic algorithms and engineering design, John Wiley and Sons, New York.

[29] Germann TC, Kadau K, Longini IM Jr, Macken CA (2006), Mitigation strategies for pandemic influenza in the United States, PNAS, 103, pp 5935-5940.

[30] Gilbert N (2007), Agent-based models—quantitative applications in the social sciences, Sage Publications, Inc.

[31] Giovachino M, McCue B (2003), Mini-Vac model documentation, Institute of Public Research Report 10889, The CNA Corporation, Alexandria, VA.

[32] Gross D, Harris CM (1998), Fundamentals of queueing theory. John Wiley and Sons, New York.

[33] Guha S, Khuller S (1999), Greedy strikes back: Improved facility location algorithms, Journal of Algorithms, 31(1), pp 228–248.

[34] Hall RW (1991), Queueing methods for services and manufacturing, Prentice Hall, Englewood Cliffs, New Jersey.

[35] Hall RW (2002), Handbook of Transportation Science, 2nd ed. Kluwer Scientific Publishers, Norwell, Massachusetts.

[36] Hethcote HW (2000), The mathematics of infectious diseases, SIAM Review, 42(4), pp 599-653.

[37] Hochbaum DS (1982), Heuristics for the fixed cost median problem, Mathematical Programming, 22(1), pp 148–162.

[38] Hopp WJ, Spearman ML (2000), Factory physics, McGraw-Hill, New York.

[39] Huerre S (2010), Agent-based crowd simulation tool for theme park environments, Proceedings of the 23rd International Conference on Computer Animation and Social Agents.

[40] Hupert N, Mushlin AJ, Callahan MA (2002), Modeling the public health response to bioterrorism: Using discrete event simulation to design antibiotic distribution centers, Medical Decision Making, 22(5 Suppl), pp S17-25.

[41] Jain K, Mahdian M, Markakis E, Saberi A, Vazirani VV (2003), Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. Journal of the ACM, 50(6), pp 795–824.

[42] Jain K, Mahdian M, Saberi A (2002), A new greedy approach for facility location problems. Proceedings of the 34th Annual ACM Symposium on Theory of Computing, pp 731–740.

[43] Jain K, Vazirani VV (2001), Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation, Journal of the ACM, 48(2), pp 274–296.

[44] Kaplan EH (2004), Preventing second-generation infections in a smallpox bioterror attack, Epidemiology, 15(3), pp 264-270.

[45] Kaplan EH, Craft DL, Wein LM (2002), Emergency response to a smallpox attack: The case for mass vaccination, PNAS, 99(16), pp 10935-10940.

[46] Kaplan EH, Craft DL, Wein LM (2003), Analyzing bio-terror response logistics: The case of smallpox, Mathematical Biosciences, 185(1), pp 33-72.

[47] Kaufmann AF, Meltzer MI, Schmid GP (1997), The economic impact of a bioterrorist attack: Are prevention and post-attack intervention programs justifiable?, Emerging Infectious Diseases, 3(2), pp 83-94.

[48] Kelton WD, Sadowski RP, Sturrock DT (2009), Simulation with Arena, 5th edition, McGraw-Hill Science/Engineering/Math.

[49] Kenneth EH III, Culver T, Keyser J, Lin M, Manocha D (2000), Fast computation of generalized Voronoi diagrams using graphics hardware, SCG '00 Proceedings of the 16th annual symposium on Computational geometry.

[50] Korupolu MR, Plaxton CG, Rajaraman R (2000), Analysis of a local search heuristic for facility location problems, J. Algorithms, 37(1), pp 146–188.

[51] Lamarche F, Donikian S (2004), Crowd of virtual humans: a new approach for real time navigation in complex and structured environments, Computer Graphics Forum, 23(3), pp 509-518.

[52] Larkin GL, Arnold J (2003), Ethical considerations in emergency planning, preparedness, and response to acts of terrorism, Prehospital and Disaster Medicine, 18(3), pp 170-178.

[53] Latombe JC (1991), Robot motion planning, Kluwer Academic.

[54] Lee EK, Chen CH, Brown N, Handy J, Desiderio A, Lopez R, Davis B (2012), Designing guest flow and operations logistics for the dolphin tales, Interfaces—The Daniel H. Wagner Prize for Excellence in Operations Research Practice, 42(5), pp 492-506.

[55] Lee EK, Chen CH, Pietz F, Benecke B (2010), Disease propagation analysis and mitigation strategies for effective mass dispensing, AMIA 2010 Symposium Proceedings, pp 427-431.

[56] Lee EK, Chen CH, Pietz F, Benecke B (2009), Modeling and optimizing the public health infrastructure for emergency response. Interfaces—The Daniel H. Wagner Prize for Excellence in Operations Research Practice, 39(5), pp 476-490.

[57] Lee EK, Chen CH, Pietz F, Benecke B, An interactive web-based decision support system for regional medical dispensing, emergency preparedness, and bio-surveillance, *submitted*.

[58] Lee EK, Maheshwary S (2006), Systems and Methods for Emergency Treatment Response and Real-Time Staff Allocation for Bioterrorism and Infectious Disease Outbreak, Copyright License Software.

[59] Lee EK, Maheshwary S, Mason J (2005), Emergency treatment response and real-time staff allocation for bioterrorism and infectious disease outbreak. The INFORMS William Pierskalla Best Paper Award on Research Excellence in HealthCare and Management Science, Nov 2005.

[60] Lee EK, Maheshwary S, Mason J, Glisson W (2006), Large-scale dispensing for emergency response to bioterrorism and infectious disease outbreak, Interfaces Homeland Security Issue, 36(6), pp 591-607.

[61] Lee EK, Maheshwary S, Mason J, Glisson W (2006), Decision support system for mass dispensing of medications for infectious disease outbreaks and bioterrorist attacks. Annals of Operations Research, 148(1), pp 25-53.

[62] Lee EK, Pietz F, Benecke B, Mason J, Burel G (2013), Advancing public health and medical preparedness with operations research. Interfaces Franz Edelman Award Issue, 43(1), pp 79-98.

[63] Lee EK, Pietz F, Benecke B, (2011) Public Health, Emergency Response, and Medical Preparedness II: Medical Countermeasures Dispensing and Large-Scale Disaster Relief Efforts. In: JJ Cochran, ed in chief. Wiley Encyclopedia of Operations Research and Management Science, Hoboken, NJ, Wiley.

[64] Lee EK, Smalley HK, Zhang Y, Pietz F, Benecke B (2009), Facility location and multi-modality mass dispensing strategies and emergency response for biodefense and infectious disease outbreaks, International Journal on Risk Assessment and Management -- Biosecurity Assurance in a Threatening World: Challenges, Explorations, and Breakthroughs, 12(2/3/4), pp 311-351.

[65] Lee EK, Zaider M (2008), Operations research advances cancer therapeutics, Interfaces-The Franz Edelman Award for Achievement in Operations Research and the Management Sciences, 38(1), pp 5-25.

[66] Li S (2011), A 1.488-approximation algorithm for the uncapacitated facility location problem. Proceedings of the 38th International Conference on Automata, Languages and Programming, pp 77-88.

[67] Longini IM Jr, Halloran ME, Nizam A, Yang Y (2004), Containing pandemic influenza with antiviral agents, American Journal of Epidemiology, 159(7), pp 623-633.

[68] Longini IM Jr, Nizam A, Xu S, Ungchusak K, Hanshaoworakul W, Cummings DA, Halloran ME (2005), Containing pandemic influenza at the source, Science, 309, pp 1083-1087.

[69] Luke S, Cioffi-Revilla C, Panait L, Sullivan K, Balan G (2005), MASON: A multi-agent simulation environment, Simulation: Transactions of the Society for Modeling and Simulation International, 82(7), pp 517-527.

[70] Mahdian M, Markakis E, Saberi A, Vazirani VV, (2001), A greedy facility location algorithm analyzed using dual fitting. Proceedings of 5th International Workshop on Randomization and Approximation, Techniques in Computer Science, volume 2129 of Lecture Notes in Computer Science, pp 127–137.

[71] Mahdian M, Ye Y, Zhang J (2006), Approximation Algorithms for Metric Facility Location Problems, SIAM Journal on Computing, 36(2), pp 411-432.

[72] Mason J, Washington M (2003), Optimizing staff allocation in large-scale dispensing centers, Report, Centers for Disease Control and Prevention, Atlanta, GA.

[73] Meltzer M, Damon I, LeDunc JW, Millar DJ (2001), Modeling potential responses to smallpox as a bioterrorist weapon, Emerging Infectious Diseases, 7(6), pp 959-969.

[74] Mills CE, Robins JM, Lipsitch M (2004), Transmissibility of 1918 pandemic influenza, Nature, 432, pp 904-906.

[75] Minar N, Burkhart R, Langton C, Askenazi M (1996), The swarm simulation system: A toolkit for building multi-agent simulations. Santa Fe (NM): Santa Fe Institute, Report No.: 96-06-042.

[76] Nemhauser GL, Wolsey LA (1988), Integer and Combinatorial Optimization. Wiley-Interscience, New York.

[77] Newell GF (1982), Applications of queueing theory, 2nd ed, Chapman and Hall, London.

[78] Nguyen S, Rosen JM, Koop CE (2005), Emerging technologies for bioweapons defense, Studies in Health Technology and Informatics, 111, pp 356-361.

[79] Nieuwenhuisen D, Kamphuis A, Mooijekind M, Overmars MH (2004), Automatic construction of high quality roadmaps for path planning, technical report, Institute of Information and Computing Sciences, Utrecht University.

[80] Nizam A, Xu S, Ungchusak K, Hanshaoworakul W, Cummings DA, Halloran ME (2005), Containing pandemic influenza at the source, Science, 309, pp 1083-1087.

[81] Patel R, Longini IM Jr, Halloran ME (2005), Finding optimal vaccination strategies for pandemic influenza using genetic algorithms, Journal of Theoretical Biology, 234(2), pp 201-212.

[82] Qin J, Ni LL, Shi F(2012), Combined Simulated Annealing Algorithm for the Discrete Facility Location Problem, The Scientific World Journal, 2012, 7 pages.

[83] Scott S, Duncan CJ (2001), Biology of Plagues: Evidence from Historical Populations, Cambridge University Press. Cambridge. UK.

[84] Shao W, Terzopoulos D (2007), Autonomous pedestrians, Graphical Models, 69, pp 246-274.

[85] Shmoys DB, Tardos E, Aardal KI (1997), Approximation algorithms for facility location problems. Proceedings of the 29th Annual ACM Symposium on Theory of Computing, pp 265–274.

[86] Sviridenko M (2002), An improved approximation algorithm for the metric uncapacitated facility location problem, Proceedings of the 9th International IPCO Conference on Integer Programming and Combinatorial Optimization, pp 240–257.

[87] Wein LM, Craft DL, Kaplan EH (2003), Emergency response to an anthrax attack, PNAS, 100(7), pp 4346-4351.

[88] Wilensky U (1999), Netlogo, http://ccl.northwestern.edu/netlogo/.

[89] Wu JT, Riley S, Fraser C, Leung GM (2006), Reducing the impact of the next influenza pandemic using household-based public health interventions, PLoS Medicine, 3(9), pp 1532-1540.

[90] Enhancing Disaster and Medical Response: Program Helps Public Health Officials Plan and Respond More Efficiently, August 15, 2008. Georgia Tech news release.

[91] Google Maps JavaScript API V3,
http://code.google.com/apis/maps/documentation/javascript/

[92] Google Maps JavaScript API V2 ,
http://code.google.com/apis/maps/documentation/javascript/v2/

[93] ArcGIS Resource Center,
http://help.arcgis.com/en/webapi/javascript/gmaps/help/google_start.htm

[94] http://schooltree.org/high/

[95] http://www.airnav.com/airports/us

[96] http://www.assistedseniorliving.net/

[97] http://www.braintrack.com/us-colleges

[98] http://www.directoryofschools.com/high-schools/US.htm

[99] http://www.homelessshelterdirectory.org/

[100] http://www.travel-library.com/hotels/north_america/usa/list.html

[101] http://zip-code-boundaries.com/index.html

[102] http://zipskinny.com/index.php